A Manifesto for Future Generation Cloud Computing: Research Directions for the Next Decade

A Manifesto for Future Generation Cloud Computing: Research Directions for the Next Decade

RAJKUMAR BUYYA*, University of Melbourne, Australia
SATISH NARAYANA SRIRAMA†‡, University of Tartu, Estonia
GIULIANO CASALE, Imperial College London, UK
RODRIGO CALHEIROS, Western Sydney University, Australia
YOGESH SIMMHAN, Indian Institute of Science, India
BLESSON VARGHESE, Queen's University Belfast, UK
EROL GELENBE, Imperial College London, UK
BAHMAN JAVADI, Western Sydney University, Australia
LUIS MIGUEL VAQUERO, University of Bristol, UK
MARCO A. S. NETTO, IBM Research, Brazil
ADEL NADJARAN TOOSI, Monash University, Australia
MARIA ALEJANDRA RODRIGUEZ, University of Melbourne, Australia
IGNACIO M. LLORENTE, Universidad Complutense de Madrid, Spain
SABRINA DE CAPITANI DI VIMERCATI, Università degli Studi di Milano, Italy
PIERANGELA SAMARATI, Università degli Studi di Milano, Italy
DEJAN MILOJICIC, Hewlett Packard Labs, USA
CARLOS VARELA, Rensselaer Polytechnic Institute, USA
RAMI BAHSOON, University of Birmingham, UK
MARCOS DIAS DE ASSUNCAO, INRIA, France
OMER RANA, Cardiff University, UK
WANLEI ZHOU, University of Technology Sydney, Australia
HAI JIN, Huazhong University of Science and Technology, China
WOLFGANG GENTZSCH, UberCloud, USA
ALBERT ZOMAYA, University of Sydney, Australia
HAIYING SHEN, University of Virginia, USA

*Corresponding author
†Co-led this work with first author; Co-First author; Corresponding author
‡Also with University of Melbourne.

Authors' addresses: Rajkumar Buyya, University of Melbourne, Australia, rbuyya@unimelb.edu.au; Satish Narayana Srirama, University of Tartu, Estonia, srirama@ut.ee; Giuliano Casale, Imperial College London, UK; Rodrigo Calheiros, Western Sydney University, Australia; Yogesh Simmhan, Indian Institute of Science, India; Blesson Varghese, Queen's University Belfast, UK; Erol Gelenbe, Imperial College London, UK; Bahman Javadi, Western Sydney University, Australia; Luis Miguel Vaquero, University of Bristol, UK; Marco A. S. Netto, IBM Research, Brazil; Adel Nadjaran Toosi, Monash University, Australia; Maria Alejandra Rodriguez, University of Melbourne, Australia; Ignacio M. Llorente, Universidad Complutense de Madrid, Spain; Sabrina De Capitani di Vimercati, Università degli Studi di Milano, Italy; Pierangela Samarati, Università degli Studi di Milano, Italy; Dejan Milojicic, Hewlett Packard Labs, USA; Carlos Varela, Rensselaer Polytechnic Institute, USA; Rami Bahsoon, University of Birmingham, UK; Marcos Dias de Assuncao, INRIA, France; Omer Rana, Cardiff University, UK; Wanlei Zhou, University of Technology Sydney, Australia; Hai Jin, Huazhong University of Science and Technology, China; Wolfgang Gentzsch, UberCloud, USA; Albert Zomaya, University of Sydney, Australia; Haiying Shen, University of Virginia, USA.

The Cloud computing paradigm has revolutionised the computer science horizon during the past decade and has enabled the emergence of computing as the fifth utility. It has captured significant attention of academia, industries, and government bodies. Now, it has emerged as the backbone of modern economy by offering subscription-based services anytime, anywhere following a pay-as-you-go model. This has instigated (1) shorter establishment times for start-ups, (2) creation of scalable global enterprise applications, (3) better cost-to-value associativity for scientific and high performance computing applications, and (4) different invocation/execution models for pervasive and ubiquitous applications. The recent technological developments and paradigms such as serverless computing, software-defined networking, Internet of Things, and processing at network edge are creating new opportunities for Cloud computing. However, they are also posing several new challenges and creating the need for new approaches and research strategies, as well as the re-evaluation of the models that were developed to address issues such as scalability, elasticity, reliability, security, sustainability, and application models. The proposed manifesto addresses them by identifying the major open challenges in Cloud computing, emerging trends, and impact areas. It then offers research directions for the next decade, thus helping in the realisation of Future Generation Cloud Computing.

CCS Concepts: • **General and reference** → **Surveys and overviews**; • **Computer systems organization** → **Cloud computing**; • **Information systems** → *Cloud based storage*; Data centers; • **Security and privacy** → *Security services*; • **Networks** → *Cloud computing*; • **Software and its engineering** → *Cloud computing*;

Additional Key Words and Phrases: Cloud computing, scalability, sustainability, InterCloud, data management, Cloud economics, application development, Fog computing, serverless computing

ACM Reference Format:
Rajkumar Buyya, Satish Narayana Srirama, Giuliano Casale, Rodrigo Calheiros, Yogesh Simmhan, Blesson Varghese, Erol Gelenbe, Bahman Javadi, Luis Miguel Vaquero, Marco A. S. Netto, Adel Nadjaran Toosi, Maria Alejandra Rodriguez, Ignacio M. Llorente, Sabrina De Capitani di Vimercati, Pierangela Samarati, Dejan Milojicic, Carlos Varela, Rami Bahsoon, Marcos Dias de Assuncao, Omer Rana, Wanlei Zhou, Hai Jin, Wolfgang Gentzsch, Albert Zomaya, and Haiying Shen. 2017. A Manifesto for Future Generation Cloud Computing: Research Directions for the Next Decade. *ACM Comput. Surv.* xx, xx, Article xx (October 2017), 53 pages.
https://doi.org/0000001.0000001

1 INTRODUCTION

Cloud computing has shaped the way in which software and IT infrastructure are used by consumers and triggered the emergence of computing as the fifth utility [38]. Since its emergence, industry organisations, governmental institutions, and academia have embraced it and its adoption has seen a rapid growth. This paradigm has developed into the backbone of modern economy by providing on-demand access to subscription-based IT resources, resembling not only the way in which basic utility services are accessed but also the reliance of modern society on them. Cloud computing has enabled new businesses to be established in a shorter amount of time, has facilitated the expansion of enterprises across the globe, has accelerated the pace of scientific progress, and has led to the creation of various models of computation for pervasive and ubiquitous applications, among other benefits.

Up to now, there have been three main service models that have fostered the adoption of Clouds, namely Software, Platform, and Infrastructure as a Service (SaaS, PaaS, and IaaS). SaaS offers the highest level of abstraction and allows users to access applications hosted in Cloud data centres (CDC), usually over the Internet. This, for instance, has allowed businesses to access software in a flexible manner by enabling unlimited and on-demand access to a range of ready-to-use applications.

0360-0300/2017/10-ARTxx $15.00
https://doi.org/0000001.0000001

SaaS has also allowed organisations to avoid incurring in internal or direct expenses, such as license fees and IT infrastructure maintenance. PaaS is tailored for users that require more control over their IT resources and offers a framework for the creation and deployment of Cloud applications that includes features such as programming models and auto-scaling. This, for example, has allowed developers to easily create applications that benefit from the elastic Cloud resource model. Finally, IaaS offers access to computing resources, usually by leasing Virtual Machines (VMs) and storage space. This layer is not only the foundation for SaaS and PaaS, but has also been the pillar of Cloud computing. It has done so by enabling users to access the IT infrastructure they require only when they need it, to adjust the amount of resources used in a flexible way, and to pay only for what has been used, all while having a high degree of control over the resources.

1.1 Motivation and Goals of the Manifesto

Throughout the evolution of Cloud computing and its increasing adoption, not only have the aforementioned models advanced and new ones emerged, but also the technologies in which this paradigm is based (e.g., virtualization) have continued to progress. For instance, the use of novel virtualization techniques such as containers that enable improved utilisation of the physical resources and further hide the complexities of hardware is becoming increasingly widespread, even leading to a new service model being offered by providers known as Container as a Service (CaaS). There has also been a rise in the type and number of specialised Cloud services that aid industries in creating value by being easily configured to meet specific business requirements. Examples of these are emerging, easy-to-use, Cloud-based data analytics services and serverless architectures.

Another clear trend is that Clouds are becoming increasingly geographically distributed to support emerging application paradigms. For example, Cloud providers have recently started extending their infrastructure and services to include edge devices for supporting emerging paradigms such as the Internet of Things (IoT) and Fog computing. Fog computing aims at moving decision making operations as close to the data sources as possible by leveraging resources on the edge such as mobile base stations, gateways, network switches and routers, thus reducing response time and network latencies. Additionally, as a way of fulfilling increasingly complex requirements that demand the composition of multiple services and as a way of achieving reliability and improving sustainability, services spanning across multiple geographically distributed CDCs have also become more widespread.

The adoption of Cloud computing will continue to increase and support for these emerging models and services is of paramount importance. In 2016, the IDG's Cloud adoption report found that 70% of organisations have at least one of their applications deployed in the Cloud and that the numbers are growing [123]. In the same year, the IDC's (International Data Corporation) Worldwide Semiannual Public Cloud Services Spending Guide [122] reported that Cloud services were expected to grow from $70 billion in 2015 to more than $203 billion in 2020, an annual growth rate almost seven times the rate of overall IT spending growth. This extensive usage of Cloud computing in various emerging domains is posing several new challenges and is forcing us to rethink the research strategies and re-evaluate the models that were developed to address issues such as scalability, resource management, reliability, and security for the realisation of next-generation Cloud computing environments [217].

This comprehensive manifesto brings these advancements together and identifies open challenges that need to be addressed for realising the *Future Generation Cloud Computing*. Given that rapid changes in computing/IT technologies in a span of 4-5 years are common, and the focus of the manifesto is for the next decade, we envision that identified research directions get addressed and will have impact on the next two or three generations of utility-oriented Cloud computing technologies, infrastructures, and their applications' services. The manifesto first discusses major

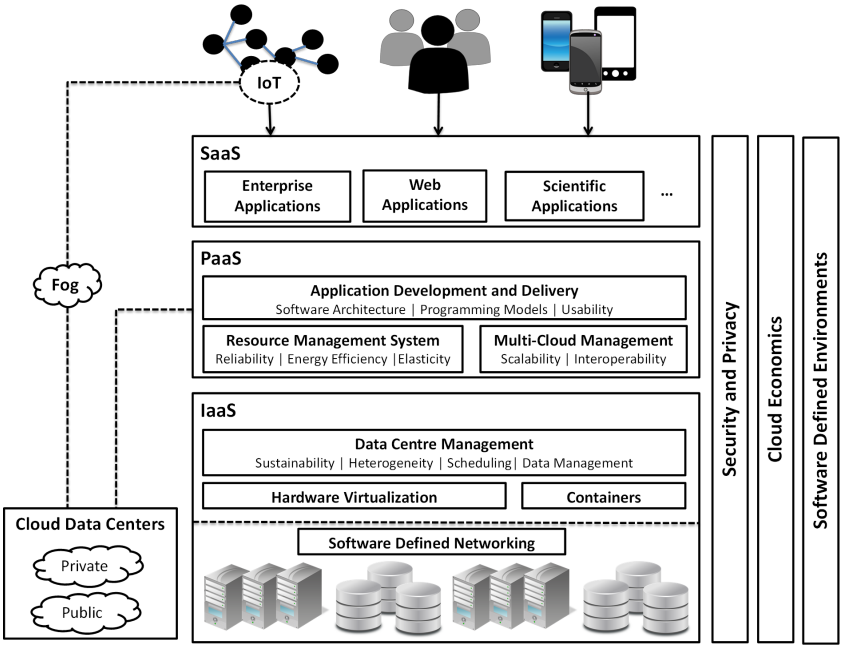

Fig. 1. Components of the Cloud computing paradigm

challenges in Cloud computing, investigates their state-of-the-art solutions, and identifies their limitations. The manifesto then discusses the emerging trends and impact areas, that further drive these Cloud computing challenges. Having identified these open issues, the manifesto then offers comprehensive future research directions in the Cloud computing horizon for the next decade. Figure 1 illustrates the main components of the Cloud computing paradigm and positions the identified trends and challenges, which are discussed further in the next sections.

The rest of the paper is organised as follows: Section 2 discusses the state-of-the-art of the challenges in Cloud computing and identifies open issues. Section 3 discusses the emerging trends and impact areas related to the Cloud computing horizon. Section 4 provides a detailed discussion about the future research directions to address the open challenges of Cloud computing. In the process, the section also mentions how the respective future research directions will be guided and influenced by the emerging trends. Section 5 provides a conclusion for the manifesto.

2 CHALLENGES: STATE-OF-THE-ART AND OPEN ISSUES

As Cloud computing became popular, it has been extensively utilised in hosting a wide variety of applications. It posed several challenges (shown within the inner ring in Figure 2) such as issues with sustainability, scalability, security, and data management among the others. Over the past decade, these challenges were systematically addressed and the state-of-the-art in Cloud computing has advanced significantly. However, there remains several issues open, as summarised in the outer ring of Figure 2. The rest of the section identifies and details the challenges in Cloud computing and their state-of-the-art, along with the limitations driving their future research.

Fig. 2. Cloud computing challenges, state-of-the-art and open issues

2.1 Scalability and Elasticity

Cloud computing differs from earlier models of distributed computing such as grids and clusters, in that it promises virtually unlimited computational resources on demand. At least two clear benefits can be obtained from this promise: first, unexpected peaks in computational demand do not entail breaking service level agreements (SLAs) due to the inability of a fixed computing infrastructure to deliver users' expected quality of service (QoS), and second, Cloud computing users do not need to make significant up-front investments in computing infrastructure but can rather grow organically as their computing needs increase and only pay for resources as needed. The first (QoS) benefit of the Cloud computing paradigm can only be realised if the infrastructure supports *scalable* services, whereby additional computational resources can be allocated, and new resources have a direct, positive impact on the performance and QoS of the hosted applications. The second

(economic) benefit can only be realised if the infrastructure supports *elastic* services, whereby allocated computational resources can *follow demand* and by dynamically growing and shrinking prevent over- and under-allocation of resources.

The research challenges associated with *scalable services* can be broken into hardware, middleware, and application levels. Cloud computing providers must embrace parallel computing *hardware* including multi-core, clusters, accelerators such as Graphics Processing Units (GPUs) [237], and non-traditional (e.g., neuromorphic and future quantum) architectures, and they need to present such heterogeneous hardware to IaaS Cloud computing users in abstractions (e.g., VMs, containers) that while providing isolation, also enable performance guarantees. At the *middleware* level, programming models and abstractions are necessary, so that PaaS Cloud computing application developers can focus on functional concerns (e.g., defining *map* and *reduce* functions) while leaving non-functional concerns (e.g., scalability, fault-tolerance) to the middleware layer [128]. At the *application* level, new generic algorithms need to be developed so that inherent scalability limitations of sequential deterministic algorithms can be overcome; these include asynchronous evolutionary algorithms, approximation algorithms, and online/incremental algorithms (see e.g., [64]). These algorithms may trade off precision or consistency for scalability and performance.

Ultimately, the scalability of the Cloud is limited by the extent to which individual components, namely compute, storage and interconnects scale. Computation has been limited by the end of scaling of both Moore's law (doubling the number of transistors every 1.5 year) and Dennard scaling ("the power use stays in proportion with area: both voltage and current scale (downward) with length"). As a consequence, the new computational units do not scale any more, nor does the power use scale. This directly influences the scaling of computation performance and cost of the Cloud. Research in new technologies, beyond CMOS (Complementary Metal-Oxide-Semiconductor), is necessary for further scaling. Similar is true for memory. DRAM (Dynamic Random-Access Memory) is limiting the cost and scaling of existing computers and new non-volatile technologies are being explored that will introduce additional scaling of load-store operating memory while reducing the power consumption. Finally, the photonic interconnects are the third pillar that enables the so called silicon photonics to propagate photonic connections into the chips improving performance, increasing scale, and reducing power consumption.

On the other hand, the research challenges associated with *elastic services* include the ability to accurately predict computational demand and performance of applications under different resource allocations [126, 202], the use of these workload and performance models in informing resource management decisions in middleware [129], and the ability of applications to scale up and down, including dynamic creation, mobility, and garbage collection of VMs, containers, and other resource abstractions [215]. While virtualization (e.g., VMs) has achieved steady maturity in terms of performance guarantees rivalling native performance for CPU-intensive applications, ease of use of containers (especially quick restarts) has led to the adoption of containers by the developers community [76]. Programming models that enable dynamic reconfiguration of applications significantly help in elasticity [214], by allowing middleware to move computations and data across Clouds, between public and private Clouds, and closer to edge resources as needed by future Cloud applications running over sensor networks such as the IoT.

In summary, scalability and elasticity provide operational capabilities to improve performance of Cloud computing applications in a cost-effective way, yet to be fully exploited. However, resource management and scheduling mechanisms need to be able to strategically use these capabilities.

2.2 Resource Management and Scheduling

The scale of modern CDCs has been rapidly growing and as of today they contain computing and storage devices in the range of tens to hundreds of thousands, hosting complex Cloud applications

and relevant data. This makes the adoption of effective resource management and scheduling policies important to achieve high scalability and operational efficiency.

Nowadays, IaaS providers mostly rely on either *static* VM provisioning policies, which allocate a fixed set of physical resources to VMs using bin-packing algorithms, or *dynamic* policies, capable of handling load variations through live VM migrations and other load balancing techniques [157]. These policies can either be reactive or proactive, and typically rely on knowledge of VM resource requirements, either user-supplied or estimated using monitoring data and forecasting.

Resource management methods are also important for PaaS and SaaS providers to help managing the type and amount of resources allocated to distributed applications, containers, web-services and micro-services. Policies available at this level include for example: 1) auto-scaling techniques, which dynamically scale up and down resources based on current and forecasted workloads; 2) resource throttling methods, to handle workload bursts, trends, smooth auto-scaling transients, or control usage of preemptible VMs (e.g., micro VMs); 3) admission control methods, to handle peak load and prioritize workloads of high-value customers; 4) service orchestration and workflow schedulers, to compose and orchestrate workloads, possibly specialised for the target domain (e.g., scientific data workflows [156]), which make decisions based on their cost-awareness and the constraint requirements of tasks; 5) multi-Cloud load balancers, to spread the load of an application across multiple CDCs.

The area of resource management and scheduling has spawned a large body of research, some recent surveys include [12, 153, 158, 194]. However, several challenges and limitations still remain. For example, existing management policies tend to be intolerant to inaccurate estimates of resource requirements, calling for studying novel trade-offs between policy optimality and its robustness to inaccurate workload information [130]. Further, demand estimation and workload prediction methods can be brittle and it remains an open question whether Machine Learning (ML) and Artificial Intelligence (AI) methods can fully address this shortcoming [41]. Another frequent issue is that resource management policies tend to focus on optimising specific metrics and resources, often lacking a systematic approach to co-existence in the same environment of multiple control loops, to ensure fair resource access across users, and to holistically optimise across layers of the Cloud stack. Novel resource management and scheduling methods for hybrid Clouds and federated Clouds also need to be devised [127]. Risks related to the interplay between security and resource management are also insufficiently addressed in current research work.

2.3 Reliability

Reliability is another critical challenge in Cloud computing environments. Data centres hosting Cloud computing consist of highly interconnected, and interdependent systems. Because of their scale, complexity and interdependencies, Cloud computing systems face a variety of reliability related threats such as hardware failures, resource missing failures, overflow failures, network failures, timeout failures and flaws in software being triggered by environmental change. Some of these failures can escalate and devastatingly impact system operation, thus causing critical failures [106]. Moreover, a cascade of failures may be triggered leading to large-scale service disruptions with far-reaching consequences [132]. As organisations are increasingly interested in adapting Cloud computing technology for applications with stringent reliability assurance and resilience requirements [191], there is an urgent demand for new ways to provision Cloud services with assured performance and resilience to deal with all types of independent and correlated failures [62]. Moreover, the mutual impact of reliability and energy efficiency of Cloud systems is one of the current research challenges [221].

Although reliability in distributed computing has been studied before [178], standard fault tolerance and reliability approaches cannot be directly applied in Cloud computing systems. The

scale and expected reliability of Cloud computing are increasingly important but hard to analyse due to the range of inter-related characteristics, e.g. their massive-scale, service sharing models, wide-area network, and heterogeneous software/hardware components. Previously, independent failures have mostly been addressed separately, however, the investigation into their interplay has been completely ignored [103]. Furthermore, since Cloud computing is typically more service-oriented rather than resource-oriented, reliability models for traditional distributed systems cannot be directly applied to Cloud computing. So, existing state-of-the-art Cloud environments lack thorough service reliability models, automatic reliability-aware service management mechanisms, and failure-aware provisioning policies.

2.4 Sustainability

Sustainability is the greatest challenge of our century, and ICT in general [89] utilises today close to 10% of all electricity consumed world-wide, resulting in a CO_2 impact that is comparable to that of air-travel. In addition to the energy consumed to operate ICT systems, we know that substantial electricity is used to manufacture electronic components, and then decommission them after the end of their useful life-time; the amount of energy consumed in this process can be 4-5 fold greater than the electricity that this equipment will consume to operate during its lifetime.

CDC deployments until recently have mainly focused on high performance and have not paid enough attention to energy consumption. Thus, today a typical CDC's energy consumption is similar to that of 25,000 households [139], while the total number of operational CDCs worldwide is 8.5 million in 2017 according to IDC. Indeed, according to Greenpeace, Cloud computing worldwide consumes more energy than most countries and only the four largest economies (USA, China, Russia, and Japan) surpass Clouds in their annual electricity usage. As the energy consumption, and the relative cost of energy in the total expenditures for the Cloud, rapidly increases, not enough research has gone into minimising the amount of energy consumed by Clouds, information systems that exploit Cloud systems, and networks [34, 177].

On the other hand, networks and the Cloud also have a huge potential to save energy in many areas such as smart cities, or to be used to optimise the mix of renewable and non-renewable energy worldwide [193]. However, the energy consumption of Clouds cannot be viewed independently of the QoS that they provide, so that both energy and QoS must be managed in conjunction. Indeed, for a given computer and network technology, reduced energy consumption is often coupled with a reduction of the QoS that users will experience. In some cases, such as critical or even life-threatening real-time needs, such as Cloud support of search and rescue operations, hospital operations or emergency management, a Cloud cannot choose to save energy in exchange for reduced QoS.

Current Cloud systems and efforts have in the past primarily focused on consolidation of VMs for minimising energy consumption of servers [24]. But other elements of CDC infrastructures, such as cooling systems (close to 35% of energy) and networks, which must be very fast and efficient, also consume significant energy that needs to be optimised by proper scheduling of the traffic flows between servers (and over high-speed networks) inside the data centre [97].

Because of multi-core architectures, novel hardware based sleep-start controls and clock speed management techniques, the power consumption of servers increasingly depends, and in a non-linear manner, on their instantaneous workload. Thus new ML-based methods have been developed to dynamically allocate tasks to multiple servers in a CDC or in the Fog [225] so that a combination of violation of SLA, which are costly to the Cloud operator and inconvenient for the end user, and other operating costs including energy consumption, are minimised. Holistic techniques must also address the QoS effect of networks such as packet delays on overall SLA, and the energy effects of networks for remote access to CDC [223]. The purpose of these methods is to provide online

automatic, or autonomic and self-aware methods to holistically manage both QoS and energy consumption of Cloud systems.

Recent work [236] has also shown that deep learning with neural networks can be effectively applied in experimental but realistic settings so that tasks are allocated to servers in a manner that optimises a prescribed performance profile that can include execution delays, response times, system throughput, and energy consumption of the CDC. Another approach that maximises the sustainability of Cloud systems and networks involves rationing the energy supply [88] so that the CDC can modulate its own energy consumption and delivered QoS in response, dynamically modifying the processors' variable clock rates as a function of the supply of energy. It has also been suggested that different sources of renewable and non-renewable energy can be mixed [90].

2.5 Heterogeneity

Public Cloud infrastructure has constantly evolved in the last decade. This is because service providers have increased their offerings while continually incorporating state-of-the-art hardware to meet customer demands and maximise performance and efficiency. This has resulted in an inherently heterogeneous Cloud with heterogeneity at three levels.

The first is at the VM level, which is due to the organisation of homogeneous (or near homogeneous; for example, same processor family) resources in multiple ways and configurations. For example, homogeneous hardware processors with N cores can be organised as VMs with any subset or multiples of N cores. The second is at the vendor level, which is due to employing resources from multiple Cloud providers with different hypervisors or software suites. This is usually seen in multi-Cloud environments [148]. The third is at the hardware architecture level, which is due to employing both CPUs and hardware accelerators, such as GPUs and Field Programmable Gate Arrays (FPGAs) [195].

The key challenges that arise due to heterogeneity in the Cloud are twofold. The first challenge is related to resource and workload management in heterogeneous environments. State-of-the-art in resource management focuses on static and dynamic VM placement and provisioning using global or local scheduling techniques that consider network parameters and energy consumption [55]. Workload management is underpinned by benchmarking techniques that are used for workload placement and scheduling techniques. Current benchmarking practices are reasonably mature for the first level of heterogeneity and are developing for the second level [134, 216]. However, significant research is still required to predict workload performance given the heterogeneity at the hardware architecture level. Despite advances, research in both heterogeneous resource management and workload management on heterogeneous resources remain fragmented since they are specific to their level of heterogeneity and do not work across the VM, vendor, and hardware architecture levels. It is still challenging to obtain a general purpose Cloud platform that integrates and manages heterogeneity at all three levels.

The second challenge is related to the development of application software that is compatible with heterogeneous resources. Currently, most accelerators require different (and sometimes vendor specific) programming languages. Software development practices for exploiting accelerators for example additionally require low level programming skills and has a significant learning curve. For example, CUDA or OpenCL are required for programming GPUs. This gap between hardware accelerators and high-level programming makes it difficult to easily adopt accelerators in Cloud software. It is recognised that abstracting hardware accelerators under middleware will reduce opportunities for optimising the source code for maximising performance. When the Cloud service offering is only the 'infrastructure', the onus is on individual developers to provide source code that is targeted to the hardware environment. However, when services, such as 'software' and 'platforms' are offered on the Cloud, the onus is not on the developer since the aim of these services

is to abstract the low-level technicalities away from the user. Therefore, it becomes necessary that the hardware is abstracted via a middleware for applications to exploit. Certainly, this comes at the expense of performance and fewer opportunities to optimise the code. Hence, there is a trade-off between performance and ease of use, when moving from VMs at the infrastructure level and on to using software and services available higher up in the computing stack. One open challenge in this area is developing software that is agnostic of the underlying hardware and can adapt based on the available hardware [136].

2.6 Interconnected Clouds

Although interconnection of Clouds was one of the earliest research problems that was identified in Cloud computing [26, 37, 184], Cloud interoperation continues to be an open issue since the field has rapidly evolved over the last half decade. Cloud providers and platforms still operate in silos, and their efforts for integration usually target their own portfolio of services. Cloud interoperation should be viewed as the capability of public Clouds, private Clouds, and other diverse systems to understand each other's system interfaces, configurations, forms of authentication and authorisation, data formats, and application initialisation and customisation [199].

Within the broader concept of interconnected Clouds, there are a number of methods that can be used to aggregate the functionalities and services of disparate Cloud providers and/or data centres. These techniques vary on who are the players that engage in the interconnections, its objectives, and the level of transparency in the aggregation of services offered to users [209].

Existing public Cloud providers offer proprietary mechanisms for interoperation that exhibit important limitations as they are not based on standards and open-source, and they do not interoperate with other providers. Although there are multiple efforts for standardisation, such as Open Grid Forum's (OGF) Open Cloud Computing Interface (OCCI), Storage Networking Industry Association's (SNIA) Cloud Data Management Interface (CDMI), Distributed Management Task Force's (DMTF) Cloud Infrastructure Management Interface (CIMI), DMTF's Open Virtualization Format (OVF), IEEE's InterCloud and National Institute of Standards and Technology's (NIST) Federated Cloud, the interfaces of existing Cloud services are not standardised and different providers use different APIs, formats and contextualization mechanisms for comparable Cloud services.

Broadly, the approaches can be classified as federated Cloud computing, if the interconnection is initiated and managed by providers (and usually transparent to users) as InterCloud or hybrid Clouds if initiated and managed by users or third parties on behalf of the users.

Federated Cloud computing is considered as the next step in the evolution of Cloud computing and an integral part of the new emerging Edge and Fog computing architectures. The federated Cloud model is gaining increasing interest in the IT market, since it can bring important benefits for companies and institutions, such as resource asset optimisation, cost savings, agile resource delivery, scalability, high availability and business continuity, and geographic dispersion [37].

In the area of InterClouds and hybrid Clouds, Moreno et al. notice that a number of approaches were proposed to provide "*the necessary mechanisms for sharing computing, storage, and networking resources*" [164]. This happens for two reasons. First, companies would like to use as much as possible of their existing in house infrastructures, for both economic and compliance reasons, and thus they should seamlessly integrate with public Cloud resources used by the company. Second, for all the workloads that are allowed to go to Clouds or for resource needs exceeding on premise capabilities, companies are seeking to offload as much of their applications as possible to the public Clouds, driven not only by the economic benefits and shared resources, but also due to the potential freedom to choose among multiple vendors on their terms.

State-of-the-art projects such as Aneka [33] have developed middleware and library solutions for integration of different resources (VMs, databases, etc.). However, the problem with such approaches

is that they need to operate in the lowest common denominator among the services offered by each provider, and this leads to suboptimal Cloud applications or support at specific service models.

Regardless of the particular Cloud interconnection pattern in place, interoperability and portability have multiple aspects and relate to a number of different components in the architecture of Cloud computing and data centres, each of which needs to be considered in its own right. These include standard interfaces, portable data formats and applications, and internationally recognised standards for service quality and security. The efficient and transparent provision, management and configuration of cross-site virtual networks to interconnect the on-premise Cloud and the external provider resources is still an important challenge that is slowing down the full adoption of this technology [121].

As Cloud adoption grows and more applications are moved to the Cloud, the need for satisfactory solutions is likely to grow. Challenges in this area concern how to go beyond the minimum common denominator of services when interoperating across providers (and thus enabling richer Cloud applications); how to coordinate authorisation, access, and billing across providers; and how to apply InterCloud solutions in the context of Fog computing and other emerging trends.

2.7 Empowering Resource-Constrained Devices

Cloud services are relevant not only for enterprise applications, but also for the resource constrained devices and their applications. With the recent innovation and development, mobile devices such as smartphones and tablets, have achieved better CPU and memory capabilities. They also have been integrated with a wide range of hardware and sensors such as camera, GPS (Global Positioning System), accelerometer etc. In addition, with the advances in 4G, 5G, and ubiquitous WiFi, the devices have achieved significantly higher data transmission rates. This progress has led to the usage of these devices in a variety of applications such as mobile commerce, mobile social networking and location based services. While the advances in the mobiles are significant and they are also being used as service providers, they still have limited battery life and when compared to desktops have limited CPU, memory and storage capacities, for hosting/executing resource-intensive tasks/applications. These limitations can be addressed by harnessing external Cloud resources, which led to the emergence of Mobile Cloud paradigm.

Mobile Cloud has been studied extensively during the past years [67] and the research mainly focused at two of its binding models, the *task delegation* and the *mobile code offloading* [78]. With the task delegation approach, the mobile invokes web services from multiple Cloud providers, and thus faces issues such as Cloud interoperability and requirement of platform specific API. Task delegation is accomplished with the help of middlewares [78]. Mobile code offloading, on the other hand, profiles and partitions the applications, and the resource-intensive methods/operations are identified and offloaded to surrogate Cloud instances (Cloudlets/swarmlets). Typical research challenges here include developing the ideal offloading approach, identifying the resource-intensive methods, and studying ideal decision mechanisms considering both the device context (e.g. battery level and network connectivity) and Cloud context (e.g. current load on the Cloud surrogates) [77, 239]. While applications based on task delegation are common, mobile code offloading is still facing adaptability challenges [77].

Correspondingly, IoT has evolved as *"web 4.0 and beyond"* and *"Industry 4.0"*, where physical objects with sensing and actuator capabilities, along with the participating individuals, are connected and communicate over the Internet [201]. There are predictions that billions of such devices/*things* will be connected using advances in building innovative physical objects and communication protocols [73]. Cloud primarily helps IoT by providing resources for the storage and distributed processing of the acquired sensor data, in different scenarios. While this *Cloud-centric IoT* model [105, 201] is interesting, it ends up with inherent challenges such as network latencies for scenarios

with sub-second response requirements. An additional aspect that arises with IoT devices is their substantial energy consumption, which can be mitigated by the use of renewable energy [90], but this in turn raises the issue of QoS as the renewable energy sources are generally sporadic. To address these issues and to realise the IoT scenarios, Fog computing is emerging as a new trend to bring computing and system supervisory activities closer to the IoT devices themselves, which is discussed in detail in Section 3.2. Fog computing mainly brings several advantages to IoT devices, such as security for edge devices, cognition of situations, agility of deployment, ultra-low latency, and efficiency on cost and performance, which are all critical challenges in the IoT environments.

2.8 Security and Privacy

Security is a major concern in ICT systems and Cloud computing is no exception. Here, we provide an overview of the existing solutions addressing problems related to the secure and private management of data and computations in the Cloud (confidentiality, integrity, and availability) along with some observations on their limitations and challenges that still need to be addressed.

With respect to the confidentiality, existing solutions typically encrypt the data before storing them at external Cloud providers [110]. Encryption, however, limits the support of query evaluation at the provider side. Solutions addressing this problem include the definition of *indexes*, which enable (partial) query evaluation at the provider side without the need to decrypt data, and the use of *encryption techniques* that support the execution of operations or the evaluation of conditions directly over encrypted data. Indexes are metadata that preserve some of the properties of the attributes on which they have been defined and can then be used for query evaluation (e.g., [5, 57, 110]). The definition of indexes must balance precision and privacy: precise indexes offer efficient query execution, but may lead to improper exposure of confidential information. Encryption techniques supporting the execution of operations on encrypted data without decryption are, for example, Order Preserving Encryption (OPE) that allows the evaluation of range conditions (e.g., [4, 222]), and fully (or partial) homomorphic encryption that allows the evaluation of arbitrarily complex functions on encrypted data (e.g., [31, 99, 100]). Taking these encryption techniques as basic building blocks, some encrypted database systems have been developed (e.g., [11, 180]), which support SQL queries over encrypted data.

Another interesting problem related to the confidentiality and privacy of data arises when considering modern Cloud-based applications (e.g., applications for accurate social services, better healthcare, detecting fraud, and national security) that explore data over multiple data sources with cross-domain knowledge. A major challenge of such applications is to preserve privacy, as data mining tools with cross-domain knowledge can reveal more personal information than anticipated, therefore prohibiting organisations to share their data. A research challenge is the design of theoretical models and practical mechanisms to preserve privacy for cross-domain knowledge [241]. Furthermore, the data collected and stored in the Cloud (e.g., data about the techniques, incentives, internal communication structures, and behaviours of attackers) can be used to verify and evaluate new theory and technical methods (e.g., [114, 207]). A current booming trend is to use ML methods in information security and privacy to analyse Big Data for threat analysis, attack intelligence, virus propagation, and data correlations [113].

Many approaches protecting the confidentiality of data rely on the implicit assumption that any authorised user, who knows the decryption key, can access the whole data content. However, in many situations there is the need of supporting *selective visibility* for *different users*. Works addressing this problem are based on *selective encryption* and on *attribute-based encryption* (ABE) [220]. Policy updates are supported, for example, by *over-encryption*, which however requires the help of the Cloud provider, and by the *Mix&Slice* approach [16], which departs from the support of the Cloud provider and uses different rounds of encryption to provide complete mixing of the resource.

The problem of selective sharing has been considered also in scenarios where different parties cooperate for sharing data and to perform distributed computations.

Alternative solutions to encryption have been adopted when associations among the data are more sensitive than the data themselves [51]. Such solutions split data in different fragments stored at different servers or guaranteed to be non linkable. They support only certain types of sensitive constraints and queries and the computational complexity for retrieving data increases.

While all solutions described above successfully provide efficient and selective access to outsourced data, they are exposed to attacks exploiting frequency of accesses to violate data and users privacy. This problem has been addressed by *Private Information Retrieval* (PIR) techniques, which operate on publicly available data, and, more recently by *privacy-preserving indexing techniques* based on, for example, Oblivious RAM, B-tree data structures, and binary search tree [65]. This field is still in its infancy and the development of practical solutions is an open problem.

With respect to the integrity, different techniques such as digital signatures, Provable Data Possession, Proof Of Retrievability, let detecting unauthorised modifications of data stored at an external Cloud provider. Verifying the integrity of stored data by its owner and authorised users is, however, only one of the aspects of integrity. When data can change dynamically, possibly by multiple writers, and queries need to be supported, several additional problems have to be addressed. Researchers have investigated the use of authenticated data structures (*deterministic* approaches) or insertion of integrity checks (*probabilistic* approaches) [60] to verify the correctness, completeness, and freshness of a computation. Both deterministic and probabilistic approaches can represent promising directions but are limited in their applicability and integrity guarantees provided.

With respect to the availability, some proposals have focused on the problem of how a user can select the services offered by a Cloud provider that match user's security and privacy requirements [58]. Typically, the expected behaviours of Cloud providers are defined by SLAs stipulated between a user and the Cloud provider itself. Recent proposals have addressed the problem of exploring possible dependencies among different characteristics of the services offered by Cloud providers [61]. These proposals represent only a first step in the definition of a comprehensive framework that allows users to select the Cloud provider that best fits their needs, and verifies that providers offer services fully compliant with the signed contract.

Hardware-based techniques have also been adopted to guarantee the proper protection of sensitive data in the Cloud. Some of the most notable solutions include the *ARM TrustZone* and the *Intel Software Guard Extensions* (SGX) technology. ARM TrustZone introduces several hardware-assisted security extensions to ARM processor cores and on-chip peripherals. The platform is then split into a "secure world" and a "normal world", each of which has different privileges and a controlled communication interface. The Intel SGX technology supports the creation of trusted execution environments, called *enclaves*, where sensitive data can be stored and processed.

Advanced *cyberattacks* in the Cloud domain represent a serious threat that may affect the confidentiality, integrity, and availability of data and computations. In particular, Advanced Persistent Threats (APTs) deserves a particular mention. This is an emerging class of cyberattacks that are goal-oriented, highly-targeted, well-organised, well-funded, technically-advanced, stealthy, and persistent. The notorious Stuxnet, Flame, and Red October are some examples of APTs. APTs poses a severe threat to the Cloud computing domain, as APTs have special characteristics that can disable the existing defence mechanisms of Cloud computing such as antivirus, firewall, intrusion detection, and antivirus [233]. Indeed, APT-based cyber breach instances and cybercrime activities have recently been on the rise, and it has been predicted that a 50% increase in security budgets will be observed to rapidly detect and respond to them [32]. In this context, enhancing the technical

levels of cyber defence only is far from being enough [82]. To mitigate the loss caused by APTs, practicable APT-targeting security solutions must be developed.

2.9 Economics of Cloud Computing

Research themes in Cloud economics have centred on a number of key aspects over recent years: (1) pricing of Cloud services – i.e. how a Cloud provider should determine and differentiate between different capabilities they offer, at different price bands and durations (e.g. micro, mini, large VM instances); (2) brokerage mechanisms that enable a user to dynamically search for Cloud services that match a given profile within a predefined budget; (3) monitoring to determine if user requirements are being met, and identifying penalty (often financial) that must be paid by a Cloud provider if values associated with pre-agreed metrics have been violated. The last of these has seen considerable work in the specification and implementation of SLAs, including implementation of specifications such as WS-Agreement.

SLA is traditionally a business concept, as it specifies contractual financial agreements between parties who engage in business activities. Faniyi and Bahsoon [75] observed that up to three SLA parameters (performance, memory, and CPU cycle) are often used. SLA management also relates to the supply and demand of computational resources, instances and services [30, 36]. A related area of *policy-based approaches* is also studied extensively [40]. Policy-based approaches are effective when resource adaptation scenarios are limited in number. As the number of encoded policies grow, these approaches can be difficult to scale. Various optimisation strategies have been used to enable SLA and policy-based resource enforcement.

Another related aspect in Cloud economics has been an understanding of how an organisation migrates current in-house or externally hosted infrastructure to Cloud providers, involving the migration of an in-house IT department to a Cloud provider. Migration of existing services needs to take account of both social and economic aspects of how Cloud services are provisioned and subsequently used, and risk associated with uptime and availability of often business critical capability. Migrating systems management capabilities outside an organisation also has an influence on what skills need to be retained within an organisation. According to a survey by RightScale [229], IT departments may now be acting as potential brokers for services that are hosted, externally within a data centre. Systems management personnel may now be acting as intermediaries between internal user requests and technical staff at the CDC, whilst some companies may fully rely instead on technical staff at the data centre, completely removing the need for local personnel. This would indicate that small companies, in particular, may not need to retain IT skills for systems management and administration, instead relying on pre-agreed SLAs with CDCs. This has already changed the landscape of the potential skills base in IT companies. Many Universities also make use of Microsoft Office 365 for managing email, an activity that was closely guarded and managed by their Information Services/IT departments in the past.

The above context has also been motivated with interest in new implementation technologies such as sub-second billing made possible through container-based deployments, often also referred to as "serverless computing", such as in Google "functions", AWS Lambda, amongst others. Serverless computing is discussed further in Section 3.4.

Licensing is another economics-related issue, which can include annual or perpetual licensing. These can be restrictive for Cloud resources (e.g. not on-demand, limited number of cores, etc.) when dealing with the demands of large business and engineering simulations for physics, manufacturing, etc. Independent Software Vendors (ISVs) such as ANSYS, Dassault, Siemens, and COMSOL are currently investigating or already have more suitable licensing models for the Cloud, such as BYOL (bring your own license), or credits/tokens/elastic units, or fully on-demand.

Another challenge in Cloud economics is related to choosing the right Cloud provider. Comparing offerings between different Cloud providers is time consuming and often challenging, as providers do not use the same terminology when offering computational and storage resources, making a like-for-like comparison difficult. A number of commercial and research grade platforms have been proposed to investigate benefit/limits of Cloud selection, such as RightScale PlanForCloud, CloudMarketMaker [133], pricing tools from particular providers (e.g. Amazon Cost Calculator, and SMI (Service Measurement Index) for ranking Cloud services [87]. Such platforms focus on what the user requires and hide the internal details of the Cloud provider's resource specifications and pricing models. In addition, marketplace models are also studied where users purchase services from SaaS providers that in turn procure computing resources from either PaaS or IaaS providers [9].

2.10 Application Development and Delivery

Cloud computing empowers application developers with the ability to programmatically control infrastructure resources and platforms. Several benefits have emerged from this feature, such as the ability to couple the application with auto-scaling controllers and to embed in the code advanced self-* mechanisms for organising, healing, optimising, and securing the Cloud application at runtime.

A key benefit of *resource* programmability is a looser boundary between development and operations, which results in the ability to accelerate the delivery of changes to the production environment. To support this feature, a variety of agile delivery tools and model-based orchestration languages (e.g., Terraform and OASIS TOSCA) are increasingly adopted in Cloud application delivery pipelines and DevOps methodologies [21]. These tools help automating lifecycle management, including continuous delivery and continuous integration, application and platform configuration, and testing.

In terms of *platform* programmability, separation of concerns has helped in tackling the complexity of software development for the Cloud and runtime management. For example, MapReduce enables application developers to specify functional components of their application, namely *map* and *reduce* functions on their data; while enabling the middleware layers to deal with non-functional concerns, such as parallelisation, data locality optimisation, and fault-tolerance. Several other programming models have emerged and are currently being investigated, to cope with the increasing heterogeneity of Cloud platforms. For example, in Edge computing, the effort to split applications relies on the developers [49]. Recent efforts in this area are also not yet fully automated [137]. Problems of this kind can be seen in many situations. Even though it is expected that there will be a wide variety and large number of edge devices and applications, there is a shortage of application delivery frameworks and programming models to deliver software spanning both the Edge and the CDC, to enable the use of heterogeneous hardware within Cloud applications, and to facilitate InterClouds operation.

Besides supporting and amplifying the above trends, an important research challenge is application evolution. Accelerated and continuous delivery may foster a short-term view of the application evolution, with a shift towards reacting to quality problems arising in production rather than avoiding them through careful design. This is in contrast with traditional approaches, where the application is carefully designed and tested to be as bug-free as possible prior to release. However, the traditional model requires more time between releases and thus it is less agile than continuous delivery methods. There is still a shortage of research in Cloud software engineering methods to combine the strengths of these two delivery approaches. For example, continuous acquisition of performance and reliability data across Cloud application releases may be used to better inform application evolution, to automate the process of identifying design anti-patterns, and to explore

what-if scenario during testing of new features. Holistic methods to implement this vision need to be systematically investigated over the coming years.

2.11 Data Management

One of the key selling points of Cloud computing is the availability of affordable, reliable and elastic storage, that is collocated with the computational infrastructure. This offers a diverse suite of storage services to meet most common enterprise needs while leaving the management and hardware costs to the IaaS service provider. They also offer reliability and availability through multiple copies that are maintained transparently, along with disaster recovery with storage that can be replicated in different regions. A number of storage abstractions are also offered to suit a particular application's needs, with the ability to acquire just the necessary quantity and pay for it. *Object-based storage* (Amazon Simple Storage Service (S3), Azure File), *block storage services* (Azure Blob, Amazon Elastic Block Store (EBS)) of a disk volume, and *logical HDD* (Hard Disk Drive) *and SSD* (Solid-state Drive) disks that can be attached to VMs are common ones. Besides these, higher level data platforms such as NoSQL columnar databases, relational SQL databases and publish-subscribe message queues are available as well.

At the same time, there has been a proliferation of Big Data platforms [146] running on distributed VM's collocated with the data storage in the data centre. The initial focus has been on batch processing and NoSQL query platforms that can handle large data volumes from web and enterprise workloads, such as Apache Hadoop, Spark and HBase. However, fast data platforms for distributed stream processing such as Apache Storm, Heron, and Apex have grown to support data from sensors and Internet-connected devices. PaaS offerings such as Amazon ElasticMR, Kinesis, Azure HDInsight and Google Dataflow are available as well.

While there has been an explosion in the data availability over the last decade, and along with the ability to store and process them on Clouds, many challenges still remain. Services for data storage have not been adequately supported by services for managing their metadata that allows data to be located and used effectively [165]. Data security and privacy remain a concern (discussed further in Section 2.8), with regulatory compliance being increasingly imposed by various governments (such as the recent EU *General Data Protection Regulation (GDPR) and US CLOUD Act*), as well as leakages due to poor data protection by users. Data is increasingly being sourced from the edge of the network as IoT device deployment grows, and the latency of wide area networks inhibits their low-latency processing. Edge and Fog computing may hold promise in this respect [219].

Even within the data centre, network latencies and bandwidth between VMs, and from VM to storage can be variable, causing bottlenecks for latency-sensitive stream processing and bandwidth-sensitive batch processing platforms. Solutions such as Software Defined Networking (SDN) and Network Functions Virtualization (NFV), which can provide mechanisms required for allocating network capacity for certain data flows both within and across data centres with certain computing operations been performed in-network, are needed [150]. Better collocation guarantees of VMs and data storage may be required as well.

There is also increasing realisation that a lambda architecture that can process both data at rest and data at motion together is essential [142]. Big Data platforms such as Apache Flink and Spark Streaming are starting to offer early solutions but further investigation is required [240]. Big Data platforms also have limited support for automated scaling out and in on elastic Clouds, and this feature is important for long-running streaming applications with dynamic workloads [145]. While the resource management approaches discussed above can help, these are yet to be actively integrated within Big Data platforms. Fine-grained per-minute and per-second billing along with faster VM acquisition time, possibly using containers, can help shape the resource acquisition better. In addition, composing applications using serverless computing such as AWS Lambda and

Azure Functions has been growing rapidly [14]. These stateless functions can off-load the resource allocation and scaling to the Cloud platform provider while relying on external state by distributed object management services like Memcached or storage services like S3.

2.12 Networking

Cloud data centres are the backbone of Cloud services where application components reside and where service logic takes place for both internal and external users. Successful delivery of Cloud services requires many levels of communication happening within and across data centres. Ensuring that this communication occurs securely, seamlessly, efficiently and in a scalable manner is a vital role of the network that ties all the service components together.

During the last decade, there has been many network-based innovations and research that have explicitly explored Cloud networking. For example, technologies such as SDN and NFV intended to build agile, flexible, and programmable computer networks to reduce both capital and operational expenditure for Cloud providers. In Section 3.5 SDN and NFV are further discussed. Likewise, scaling limitations as well as the need for a flat address space and over subscription of servers also have prompted many recent advances in the network architecture such as VL2 [104], PortLand [172], and BCube [107] for the CDCs. Despite all these advances, there are still many networking challenges that need to be addressed.

One of the main concerns of today's CDCs is their high energy consumption. Nevertheless, the general practice in many data centres is to leave all networking devices always on [116]. In addition, unlike computing servers, the majority of network elements such as switches, hubs, and routers are not designed to be energy proportional and things such as, sleeping during no traffic and adaptation of link rate during low traffic periods, are not a native part of the hardware [154]. Therefore, the design and implementation of methodologies and technologies to reduce network energy consumption and make it proportional to the load remain as open challenges.

Another challenge with CDC networks is related to providing guaranteed QoS. The SLAs of today's Clouds are mostly centred on computation and storage [108]. No abstraction or mechanism enforcing the performance isolation and hence no SLAs beyond best effort is available to capture the network performance requirements such as delay and bandwidth guarantees. Within the data centre infrastructure, Guo et al. [108] propose a network abstraction layer called VDC which works based on a source routing technique to provide bandwidth guarantees for VMs. Yet, their method does not provide any network delays guarantee. This challenge becomes even more pressing, when network connectivity must be provided over geographically distributed resources, for example, deployment of a "virtual cluster" spanning resources on a hybrid Cloud environment. Even though the network connectivity problem involving resources in multiple sites can be addressed using network virtualization technologies, providing performance guarantees for such networks as it traverses over the public Internet raises many significant challenges that require special consideration [209]. The primary challenge in this regard is that cloud providers do not have privileged access to the core Internet equipment as they do in their own data centres. Therefore, cloud providers' flexibility regarding routing and traffic engineering is limited to a large extent. Moreover, the performance of public network such as the Internet is much more unpredictable and changeable compared to the dedicate network of data centres which makes it more difficult to provide guaranteed performance requirements. Traditional WAN approaches such as Multi-Protocol Label Switching (MPLS) for traffic engineering in such networks are also inefficient in terms of bandwidth usage and handling latency-sensitive traffic due to lack of global view of the network [119]. This is one of the main reasons that companies such as Google invested on its own dedicated network infrastructures to connect its data centres across the globe [131].

In addition, Cloud networking is not a trivial task and modern CDCs face similar challenges to building the Internet due to their size [15]. The highly virtualized environment of a CDC is also posing issues that have always existed within network apart from new challenges of these multi-tenant platforms. For example in terms of scalability, VLANs (Virtual Local Area Network) are a simple example. At present, VLANs are theoretically limited to 4,096 segments. Thus, the scale is limited to approximately 4,000 tenants in a multitenant environment. VXLAN offers encapsulation methods to address the limited number of VLANs. However, it is limited in multicasting, and supports Layer 2 only within the logical network. IPv4 is another example, where some Cloud providers such as Microsoft Azure admitted that they ran out of addresses. To overcome this issue the transition to the impending IPv6 adoption must be accelerated. This requirement means that the need for network technologies offering high performance, robustness, reliability, flexibility, scalability, and security never ends [15].

2.13 Usability

The Human Computer Interface and Distributed Systems communities are still far from one another. Cloud computing, in particular, would benefit from a closer alignment of these two communities. Although much effort has happened on resource management and back-end related issues, usability is a key aspect to reduce costs of organisations exploring Cloud services and infrastructure. This reduction is possible, mainly due to labour related expenses as users can have better quality of service and enhance their productivity. The usability of Cloud [74] has already been identified as a key concern by NIST as described in their Cloud Usability Framework [203], which highlights five aspects: capable, personal, reliable, secure, and valuable. Capable is related to meeting Cloud consumers expectations with regard to Cloud service capabilities. Personal aims at allowing users and organizations to change the look and feel of user interfaces and to customise service functionalities. Reliable, secure, and valuable are aspects related to having a system that performs its functions under state conditions, safely/protected, and that returns value to users respectively. Coupa's white paper [54] on usability of Cloud applications also explores similar aspects, highlighting the importance of usability when offering services in the Internet.

For usability, current efforts in Cloud have mostly focused on encapsulating complex services into APIs to be easily consumed by users. One area where this is clearly visible is High Performance Computing (HPC) Cloud [171]. Researchers have been creating services to expose HPC applications to simplify their consumptions [50, 120]. These applications are not only encapsulated as services, but also receive Web portals to specify application parameters and manage input and output files.

Another direction related to usability of Cloud that got traction in the last years is DevOps [18, 182]. Its goal is to integrate development (Dev) and operations (Ops) thus aiding faster software delivery (as also discussed in Sections 2.10 and 4.10). DevOps has improved the productivity of developers and operators when creating and deploying solutions in Cloud environments. It is relevant not only to build new solutions in the Cloud but also to simplify the migration of legacy software from on-premise environments to multi-tenancy elastic Cloud services.

3 EMERGING TRENDS AND IMPACT AREAS

As Cloud computing and relevant research matured over the years, it led to several advancements in the underlying technologies such as containers and software defined networks. These developments in turn have led to several emerging trends in Cloud computing such as Fog computing, serverless computing, and software defined computing. In addition to them, other emerging trends in ICT such as Big Data, machine/deep learning, and blockchain technology also have started influencing the Cloud computing research and have offered wide opportunities to deal with the open issues in

Cloud-related challenges. Here we discuss the emerging trends and impact areas relevant in the Cloud horizon.

3.1 Containers

With the birth of Docker [163], container technologies have aroused wide interest in both academia and industry [196]. Containers provide a lightweight environment for the deployment applications; they are stand-alone, self-contained units that package software and its dependencies together. Similar to VMs, containers enable the resources of a single compute node to be shared by enabling applications to run as isolated user space processes.

Containers rely on modern Linux operating systems' kernel facilities such as cgroups, LXC (Linux containers) and libcontainer. Docker uses Linux kernel's cgroups and namespaces to run independent "containers" within a physical machine. Control Groups (cgroups) provide isolation of resources such as CPU, memory, block I/O and network. On the other hand, namespaces isolate an application's view of the operating environment, that includes process trees, network, user IDs and mounted file systems. Docker contains the libcontainer library as a container reference implementation. By packing the application and related dependencies into a Docker image, Docker simplifies the deployment of the application and improves the development efficiency.

More and more Internet companies are adopting this technology and containers have become the de-facto standard for creating, publishing, and running applications. This increased demand has led for instance to the emergence of CaaS (*container as a service*), a model derived from the traditional Cloud computing [185]. An example of this type of service is UberCloud [2, 101]; a platform offering application containers and their execution for a variety of engineering simulations.

The increase in popularity of containers may be attributed to two main features. First, they start up very quickly and their launching time is less than a second. Second, containers have small memory footprint and consume a very small amount of resources. Compared with VMs, using containers not only improves the performance of applications, but also allows the host to support more instances simultaneously.

Despite these advantages, there are still drawbacks and challenges that need to be addressed. First, due to the sharing of the kernel, the isolation and security of containers is weaker than in VMs [232], which stimulates much interest and enthusiasm of researchers. There are two promising solutions to this problem. One is to leverage new hardware features, such as the trusted execution support of Intel SGX [13]. The other one is to use Unikernel, which is a kind of library operating system [3]. Second, trying to optimise the performance of containers is an everlasting theme. For example, to accelerate the container start-up, Slack is proposed to optimise the storage driver [115]. Last but not least, the management of container clusters based on users' QoS requirements is attracting significant attention. Systems for container cluster management such as Kubernetes [144], Mesos [118] and Swarm [68] are emerging as the core software of the Cloud computing platform.

3.2 Fog Computing

The Fog is an extension to the traditional Cloud computing model in that the edge of the network is included in the computing ecosystem to facilitate decision making as close as possible to the data source [29, 86, 213]. The vision of Fog computing is three fold. First, to enable general purpose computing on traffic routing nodes, such as mobile base stations, gateways and routers. Second, to add compute capabilities to traffic routing nodes so as to process data as it is transmitted between user devices and a CDC. Third, to use a combination of the former.

There are a number of benefits in using such a compute model. For example, latencies between users and servers can be reduced. Moreover, location awareness can be taken into account for geo-distributed computing on edge nodes. The Fog model inherently lends itself to improving the

QoS of streaming and real-time applications. Additionally, mobility can be seamlessly supported, wireless access between user devices and compute servers can be enabled and scalable control systems can be orchestrated. These benefits make it an appropriate solution for the upcoming IoT class of applications [59, 219, 227].

Edge and Fog computing are normally used interchangeably, however, they are slightly different, both paradigms rely on local processing power near data sources. In Edge computing, the processing power is given to the IoT device itself, while in the Fog computing, computing nodes (e.g., Dockers and VMs) are placed very close the source of data. The Edge computing paradigm depends on how IoT devices can be programmed to interact with each other and run user defined codes. Unfortunately, standard APIs that provide such functionality are not fully adopted by current IoT sensors/actuators, and thus Fog computing seems to be the only viable/generic solutions to date [168].

The Fog would offer a full-stack of IaaS, PaaS, and SaaS resources, albeit not to the full extent as a CDC. Given that a major benefit of the Fog is its closer network proximity to the consumers of the services to reduce latency, it is anticipated that there will be a few Fog data centres per city. But as yet, the business model is evolving and possible locations for Fog resources range from a local coffee shop to mobile cell towers (as in Mobile Edge computing [228]). Additionally, infrastructure provided by traditional private Cloud and independent Fog providers may be employed [42]. Economics related research challenges and opportunities for Fog computing are discussed further in Section 4.9. Although the concept of Mobile Edge computing is similar to the premise of Fog computing, it is based on the mobile cellular network and does not extend to other traffic routing nodes along the path data travels between the user and the CDC.

Advantages of Fog computing include the vertical scaling of applications across different computing tiers. This allows for example, pre-processing the data contained in packets so that value is added to the data and only essential traffic is transmitted to a CDC. Workloads can be (1) decomposed on CDCs and offloaded on to edge nodes, (2) migrated from a collection of user devices on to edge nodes, or (3) aggregated from multiple sensors or devices on an edge node. In the Fog layer, workloads may be deployed via containers in lieu of VMs that require more resources [137, 152].

Cloud vendors have started to use edge locations to deliver security services (AWS Shield, Web Application Firewall Service) closer to users or to modify network traffic (e.g. Lambda@Edge). Cloud providers are also asking customers to deploy on-premise storage and compute capabilities working with the same APIs as the ones they use in their Cloud infrastructure. These have made it possible to deliver the advantages of Fog architectures to the end users. For instance, in Intensive Care Units, in order to guarantee uninterrupted care when faced with a major IT outage, or to bring storage and computing capabilities to poorly connected areas (e.g. AWS Snowball Edge for the US Department of Defense).

Other applications that can benefit from the Fog include smart city and IoT applications that are fast growing. Here, multi-dimensional data, such as text, audio and video are captured from urban and social sensors, and deep-learning models may be trained and perform inferencing to drive real-time decisions such as traffic signalling. Autonomous vehicles such as driverless cars and drones can also benefit from the processing capabilities offered by the Fog, well beyond what is hosted in the vehicle. The Fog can also offer computing and data archival capabilities. Immersive environments such as MMORPG gaming, 3D environment such as HoloLens and Google Glass, and even robotic surgery can benefit from GPGPUs that may be hosted on the Fog.

Many works such as Shi and Dustdar [192], Varghese et al. [218], Chang et al. [42] and Garcia Lopez et al. [86] have highlighted several challenges in Edge/Fog computing. Two prominent challenges that need to be addressed to enhance utility of Fog computing are mentioned here. First, tackling the complex management issues related to multi-party SLAs. To this end, as a first step

responsibilities of all parties will need to be articulated. This will be essential for developing a unified and interoperable platform for management since Edge nodes are likely to be owned by different organisations. The EdgeX Foundry [208] project aims to tackle some of these challenges. Second, given the possibility of multiple node interactions between a user device and CDC, security will need to be enhanced and privacy issues will need to be debated and addressed [205]. The Open Fog consortium [175] is a first step in this direction.

3.3 Big Data

There is a rapid escalation in the generation of streaming data from physical and crowd-sourced sensors as deployments of IoT, Cyber Physical Systems (CPS) [230], and micro-messaging social networks such as Twitter. This quantity is bound to grow many-fold, and may dwarf the size of data present on the public WWW, enterprises and mobile Clouds. Fast data platforms to deal with data velocity may usurp the current focus on data volume.

This has also seen the rise of in-memory and stream computation platforms such as Spark Streaming, Flink and Kafka that process the data in-memory as events or micro-batches and over the network rather than write to disk like Hadoop [238]. This offers a faster response for continuously arriving data, while also balancing throughput. This may put pressure on memory allocation for VMs, with SSD's playing a greater role in the storage hierarchy.

We are also seeing data acquisition at the edge by IoT and Smart City applications with an inherent feedback loop back to the edge. Video data from millions of cameras from city surveillance, self-driving cars, and drones at the edge is also poised to grow [188]. This makes latency and bandwidth between Edge and Cloud a constraint if purely performing analytics on the Cloud. Edge/Fog computing is starting to complement Cloud computing as a first-class platform, with Cloud providers already offering SDK's to make this easier from user-managed edge devices. While smartphones have already propagated mobile Clouds where applications cooperatively work with Cloud services, there will be a greater need to combine peer-to-peer computing on the Edge with Cloud services, possibly across data centres. This may also drive the need for more regional data centres to lower the network latency from the edge, and spur the growth of Fog computing.

Unlike structured data warehouses, the growing trend of *"Data Lakes"* encourages enterprises to put all their data into Cloud storage, such as HDFS, to allow intelligence to be mined from it [204]. However, a lack of tracking metadata describing the source and provenance of the data makes it challenging to use them, effectively forming *"data graveyards"*. Many of these datasets are also related to each other through logical relationships or by capturing physical infrastructure, though the linked nature of the datasets may not be explicitly captured in the storage model [28]. There is heightened interest in both deep learning platforms like TensorFlow to mine such large unstructured data lakes, as well as distributed graph databases like Titan and Neo4J to explore such linked data.

3.4 Serverless Computing

Serverless computing is an emerging architectural pattern that changes dramatically the way Cloud applications are designed. Unlike a traditional three-tiered Cloud application in which both the application logic and the database server reside in the Cloud, in a serverless application the business logic is moved to the client; this may be embedded in a mobile app or ran on temporarily provisioned resources during the duration of the request. This translates to the fact that a client does not need to rent resources, for example Cloud VMs for running the server of an application [8]. This computing model implicitly handles the challenges of deploying applications on a VM, such as over/under provisioning Cloud VMs for the application, balancing the workload across the resources and ensuring reliability and fault-tolerance. In this case, the actual server is made abstract,

such that properties like control, cost and flexibility, which are not conventionally considered are taken into account.

Consequently, serverless computing reduces the amount of backend code, developers need to write, and also reduces administration on Cloud resources. It appears in two forms; Backend as a Service (BaaS) and Functions as a Service (FaaS) [183]. This architecture is currently supported on platforms such as AWS Lambda, IBM OpenWhiskand Google Cloud Functions.

It is worth noting the term "serverless" may be somehow misleading: it does not mean that the application runs without servers; instead, it means that the resources used by the application are managed by the Cloud provider [19]. In BaaS, the server-side logic is replaced by different Cloud services that carry out the relevant tasks (for example, authentication, database access, messaging, etc.), whereas in FaaS ephemeral computing resources are utilised that are charged per access (rather than on the basis of time, which is typical of IaaS solutions).

FaaS poses new challenges particularly for resource management in Clouds that will need to be addressed. This is because arbitrary code (the function) will need to execute in the Cloud without any explicit specification of resources required for the operation. To make this possible, FaaS providers pose many restrictions about what functions can do and for how long they can operate [19]. For example, they enforce limits on the amount of time a function can execute, how functions can be written (enforcing stateless computations), and how the code is deployed [19]. This is restrictive in the types of applications that can make use of current FaaS models.

The above results in new challenges from a Software Engineering perspective: applications need to be redesigned to leverage the model, forcing software engineers to shift the way they design and think about the logic of their applications. Although some of these changes, for example, making applications stateless, is also desirable if other benefits from Clouds as elasticity are to be fully leveraged at application model, there are at least two other challenges that are particularly relevant to this model, namely event-based and timeout-aware application logic. The former issue arises because each function can be seen as a particular response to an event that will trigger other events in response to its execution. The latter arises because serverless offers implement time-outs in their logic, so it is important that this is taken into consideration during the design and execution of functions, and strategies to circumvent the time limit of applications need to be adopted whenever it is necessary.

A full-fledged general-purpose serverless computing model is still a vision that needs to be achieved. Upcoming research has explored applications that can benefit from serverless computing [234] and platforms that match services offered by providers [117, 159, 200]. As discussed by Hendrickson et al. [117], there are still a number of issues at the middleware layer that need to be addressed that are orthogonal to advances in the area of Cloud computing that are also necessary to better support this model. Despite these challenges, this is a promising area to be explored with significant practical and economic impact. It is predicted by Forbes that there will be a likely increase of serverless computing since a large number of 'things' will be connected to the edge and data centres [66].

3.5 Software-defined Cloud Computing

Software-defined Cloud Computing is a method for the optimisation and automation of configuration process and physical resources abstraction, by extending the concept of virtualization to all resources in a data centre including compute, storage, and network [35]. Virtualization technologies aim to mask, abstract and transparently leverage underlying resources without applications and clients having to understand physical attributes of the resource. Virtualization technologies for computing and storage resources are quite advanced to a large extent. The emerging trends in this

space are the virtualization in networking aspects of Cloud, namely Software-defined networking (SDN) and Network functions virtualization (NFV).

The main motivation for SDN, an emerging networking paradigm, is due to the demand/need for agile and cost-efficient computer networks that can also support multi-tenancy [167]. SDN aims at overcoming the limitations of traditional networks, in particular networking challenges of multi-tenant environments such as CDCs where computing, storage, and network resources must be offered in slices that are independent or isolated from one another. Early supporters of SDN were among those believing that networking equipment manufacturers were not meeting their needs particularly in terms of innovation and the development of required features of data centres. There were another group of supporters who aimed at running their network by harnessing the low-cost processing power of commodity hardware.

SDN decouples the data forwarding functions and network control plane, which enables the network to become centrally manageable and programmable [174]. This separation offers the flexibility of running some form of logically centralised network orchestration via the software called SDN controller. The SDN controller provides vendor-neutral open standards which abstract the underlying infrastructure for the applications and network services and facilitates communication between applications wishing to interact with network elements and vice versa [167]. OpenFlow [160] is the de-facto standard of SDN and is used by most of SDN Controllers as southbound APIs for communications with network elements such as switches and routers.

NFV is another trend in networking which is quickly gaining attention with more or less similar goals to SDN. The main aim of NFV is to transfer network functions such as intrusion detection, load balancing, firewalling, network address translation (NAT), domain name service (DNS), to name a few, from proprietary hardware appliances to software-based applications executing on commercial off-the-shelf (COTS) equipment. NFV intends to reduce cost and increase elasticity of network functions by building network function blocks that connect or chain together to build communication services [46]. Han et al. [112] presented a comprehensive survey of key challenges and technical requirements of NFV. Network service chaining, also known as service function chaining (SFC), is an automated process used by network operators to set up a chain of connected network services. SFC enables the assembly of the chain of virtual network functions (VNFs) in an NFV environment using instantiation of software-only services running on commodity hardware. Management and orchestration (MANO) of NFV environments is another popular research topic and a widely studied problem in the literature [161].

Apart from networking challenges, SDN and NFV can serve as building blocks of next-generation Clouds by facilitating the way challenges such as sustainability, interconnected Clouds, and security can be addressed. Heller et al. [116] conducted one of the early attempts towards sustainability of Cloud networks using OpenFlow switches and providing network energy proportionality. The main advantage of using NFV is that Cloud service providers can launch new network function services in a more agile and flexible way. In view of that, Eramo et al. [72] proposed a consolidation algorithm based on a migration policy of virtualized network function instances to reduce energy consumption. Google adopted SDN in its B4 network to interconnect its CDC with a globally-deployed software defined WAN [131]. Yan et al. [235] investigated how SDN-enabled Cloud brings us new opportunities for tackling distributed denial-of-service (DDoS) attacks in Cloud computing environments.

3.6 Blockchain

In several industries, blockchain technology [206] is becoming fundamental to accelerate and optimise transactions by increasing their level of traceability, reliability, and auditability. Blockchain consists of a distributed immutable ledger deployed in a decentralised network that relies on

cryptography to meet security constraints [210]. Different parties of a chain have the same copy of the ledger and have to agree on transactions being placed into the blockchain. Cloud computing is essential for blockchain as it can host not only the blockchain nodes, but services created to leverage this infrastructure. Cloud can encapsulate blockchain services in both PaaS and SaaS to facilitate their usage. This will involve also challenges related to scalability as these chains start to grow as technology matures. Cloud plays a key role in the widespread adoption of blockchain with its flexibility for dynamically allocating computing resources and managing storage [52]. An important component of blockchain is to serve as a platform to run analytics on transaction data, which can be mixed with data coming from other sources such as IoT, financial, and weather-related services. There are many transactions that happen outside the Cloud and blockchain will force such transactions to be moved to the Cloud, which will require data centres to handle a much larger load than they currently do—thus raising issues related to sustainability, mainly in terms of infrastructure energy consumption (see Section 4.4). Such a load will come not only for the transactions themselves, but all analytics services that will benefit from this transactional data. Therefore, the difficult aspect in the Cloud to handle blockchain services comes from the need of much more efficient infrastructure for transactions and all associated dynamic computing demand from smart contracts and analytics that emerge at different times and geographies according to the transactional flows.

Another side of blockchain and Cloud is to consider the direction where the advances in blockchain will assist Cloud computing [17, 83]. It is well known that Cloud is an important platform for collaboration and data exchange. Blockchain can assist Cloud by creating more secure and auditable transaction platform. This is essential for several industries including health, agriculture, manufacturing, and petroleum. This is tied to the importance of data for machine learning and deep learning solutions. Such data is generated by several users and companies that want to receive profit for their data to artificial intelligence services. Blockchain can be interleaved with cloud platforms to create trusted and verifiable data marketplaces. Consequently, users and companies can trade data and insights in an efficient, reliable, and auditable fashion. The challenge in this research area involves scalability, mechanisms to verify the usefulness/quality of data, and usability tools to facilitate such blockchain-aware data trading mechanisms.

3.7 Machine and Deep Learning

Due to the vast amount of data generated in the last years and the computing power increase, mainly of GPUs, AI has gained a lot of attention lately. Algorithms and models for machine learning and deep learning are relevant for Cloud computing researchers and practitioners. From one side, Cloud can benefit from machine/deep learning in order to have more optimised resource management, and on the other side, Cloud is an essential platform to host machine/deep learning services due to its pay-as-you-go model and easy access to computing resources.

In the early 2000s, autonomic computing was a subject of study to make computing systems more efficient through automation [140]. There, systems would have four major characteristics: self-configuration, self-optimisation, self-healing, and self-protection. The vision may become possible with the assistance of breakthroughs in artificial intelligence and data availability. For Cloud, this means efficient ways of managing user workloads, predictions of demands for computing power, estimations of SLA violations, better job placement decisions, among others. Simplifying the selection of Cloud instances [186] or optimising resource selection [20] are well known examples of the use of machine learning for better use of Cloud services and infrastructure. The industry has already started to deliver auto-tuning techniques for many Cloud services so that many aspects of running the application stack are delegated to the Cloud platform. For instance, Azure SQL database has auto-tuning as a built-in feature that adapts the database configuration (e.g. tweaking

and cleaning indices [140]). One difficult and relevant research direction in this area is to create reusable models from machine/deep learning solutions that can be used by several users/companies in different contexts instead of creating multiple solutions from scratch. The bottleneck is that applications/services have peculiarities that may block the direct reuse of solutions for resource optimisation from other users/companies.

Several machine learning and deep learning algorithms require large-scale computing power and external data sources, which can be cheaper and easier to acquire via Cloud than using on-premise infrastructure. This is becoming particularly relevant as technologies to train complex machine/deep learning models can now be executed in parallel at scale [48]. That is why several companies are providing AI-related services in the Cloud such as IBM Watson, Microsoft Azure Machine Learning, AWS Deep Learning AMIs, Google Cloud Machine Learning Engine, among others. Some of these Cloud services can be enhanced while users consume them. This has already delivered considerable savings for CDCs [84]. It can also streamline managed database configuration tuning [212].

We anticipate a massive adoption of auto-tuners, especially for the SaaS layer of the Cloud. We also foresee the likely advent of new automated tools for Cloud users to benefit from the experience of other users via semi-automated application builders (recommending tools of configurations other similar users have successfully employed), automated database sharders, query optimisers, or smart load balancers and service replicators. As security becomes a key concern for most corporations worldwide, new ML-based security Cloud services will help defend critical Cloud services and rapidly mutate to adapt to new fast-developing threats.

4 FUTURE RESEARCH DIRECTIONS

The Cloud computing paradigm, like the Web, the Internet, and the computer itself, has transformed the information technology landscape in its first decade of existence. However, the next decade will bring about significant new requirements, from large-scale heterogeneous IoT and sensor networks producing very large data streams to store, manage, and analyse, to energy- and cost-aware personalised computing services that must adapt to a plethora of hardware devices while optimising for multiple criteria including application-level QoS constraints and economic restrictions.

Significant research was already performed to address the Cloud computing technological and adoption challenges, and the state-of-the-art along with their limitations is discussed thoroughly in Section 2. The future research in Cloud computing should focus at addressing these limitations along with the problems hurled and opportunities presented by the latest developments in the Cloud horizon. Thus the future R&D will greatly be influenced/driven by the emerging trends discussed in Section 3. Here the manifesto provides the key future directions for the Cloud computing research, for the coming decade.

4.1 Scalability and Elasticity

Scalability and elasticity research challenges for the next decade can be decomposed into hardware, middleware, and application-level.

At the Cloud computing hardware level, an interesting research direction is special-purpose Clouds for specific functions, such as deep learning—e.g. Convolutional Neural Networks (CNNs), Multi-Layer Perceptrons (MLPs), and Long Short-Term Memory (LSTMs)—data stream analytics, and image and video pattern recognition. While these functionalities may appear to be very narrow, they can be deployed for a spectrum of applications and their usage is increasingly growing. There are numerous examples at control points at airports, social network mining, IoT sensor data analytics, smart transportation, and many other applications. Key Cloud providers are already offering accelerators and special-purpose hardware with increasing usage growth, e.g., Amazon is offering GPUs, Google has been deploying Tensor Processing Units (TPUs) [135] and Microsoft is

deploying FPGAs in the Azure Cloud [181]. As new hardware addresses scalability, Clouds need to embrace non-traditional architectures, such as neuromorphic, quantum computing, adiabatic, nanocomputing and many others (see [124]). Research needed includes developing appropriate virtualization abstractions, as well as programming abstractions enabling just-in-time compilation and optimisation for special-purpose hardware. Appropriate economic models also need to be investigated for FaaS Cloud providers (e.g., offering image and video processing as composable micro-services).

At the Cloud computing middleware level, research is required to further increase reuse of existing infrastructure, to improve speed of deployment and provisioning of hardware and networks for very large scale deployments. This includes algorithms and software stacks for reliable execution of applications with failovers to geographically remote private or hybrid Cloud sites. Research is also needed on InterClouds which will seamlessly enable computations to run on multiple public Cloud providers simultaneously. In order to support HPC applications, it will be critical to guarantee consistent performance across multiple runs even in the presence of additional Cloud users. New deployment and scheduling algorithms need to be developed to carefully match HPC applications with those that would not introduce noise in parallel execution or if not possible, to use dedicated clusters for HPC [109, 171].

To be able to address large scale communication-intensive applications, further Cloud provider investments are required to support high throughput and low latency networks [171]. The environment of these applications necessitates sophisticated mechanisms for handling multiple clients and for providing sustainable and profitable business provision. Moreover, Big Data applications are leveraging HPC capabilities and IoT, providing support for many modern applications such as smart cities [187] or industrial IoT [29]. These applications have demanding requirements in terms of (near-)real time processing of large scale of data, its intelligent analysis and then closing the loops of control.

4.2 Resource Management and Scheduling

The evolution of the Cloud in the upcoming years will lead to a new generation of research solutions for resource management and scheduling. Technology trends such as Fog will increase the level of decentralisation of the computation, leading to increased heterogeneity in the resources and platforms and also to more variability in the processed workloads. Technology trends, such as serverless computing and Edge computing, will also offer novel opportunities to reason on the trade-offs of offloading part of the application logic far from the system core, posing new questions on optimal management and scheduling. Conversely, trends such as software-defined computing and Big Data will come to maturity, expanding the enactment mechanisms and reasoning techniques available for resource management and scheduling, thus offering many outlets for novel research.

Challenges arising from decentralisation are inherently illustrated in the Fog computing domain, edge analytics (discussed further in Section 4.7) being one interesting research direction. In edge analytics, the stream-based or event-driven sensor data will be processed across the complete hierarchy of Fog topology. This will require cooperative resource management between centralised CDCs and distributed Edge computing resources for real-time processing. Such management methods should be aware of the locations and resources available to edge devices for optimal resource allocation, and should take into account device mobility, highly dynamic network topology, and privacy and security protection constraints at scale. The design of multiple co-existing control loops spanning from CDCs to the Edge is, by itself, a broad research challenge from the point of design, analysis and verification, implementation and testing. The adoption of container technology in these applications will be useful due to its small footprint and fast deployment [176].

Novel research challenges in the area of scheduling will also arise in these decentralised and heterogeneous environments. Recently proposed concepts such as multi-resource fairness [102] as well as non-conventional game theoretic methods [189], which today are primarily applied to small to medium-scale computing clusters or to define optimal economic models for the Cloud, need to be generalised and applied to large-scale heterogeneous settings comprising both CDCs and Edge. For example, mean-field games may help in addressing inherent scalability problems by helping to reason about the interaction of a large number of resources, devices and user types [189].

Serverless computing is an example of emerging research challenges in management and scheduling, such as offloading the computation far from the application core components that implement the business logic. From the end user standpoint, FaaS raises the expectation that functions will be executed within a specific time, which is challenging given that current performance is quite erratic [79] and network latency can visibly affect function response time. Moreover, given that function cost is per access, this will require novel resource management policies to decide when and to which extent rely on FaaS instead of microservices that run locally to the application.

From the FaaS provider perspective, allocation of resources needs to be optimal (neither excessive nor insufficient), and, from a user perspective, a desirable level of QoS needs to be achieved when functions are executed, determining suitable trade-offs with execution requirements, network latency, privacy and security requirements. Given that a single application backed by FaaS can lead to hundreds of hits to the Cloud in a second, an important challenge for serverless platform providers will be to optimise allocation of resources for each class of service so that revenue is optimised, while all the user FaaS QoS expectations are met. This research will require to take into consideration soft constraints on execution time of functions and proactive FaaS provisioning to avoid high latency of resource start-up to affect the performance of backed applications. Moreover, providers and consumers, both for FaaS and regular Cloud services, often have different goals and constraints, calling for novel game-theoretic approaches and market-oriented models for resource allocation and regulation of the supply and demand within the Cloud platform.

The emerging SDN paradigm exemplifies a novel trend which will extend the range of control mechanisms available for holistic management of resources. By logically centralising the network control plane, SDNs provide opportunities for more efficient management of resources located in a single administrative domain such as a CDC. SDN also facilitates joint VM and traffic consolidation, a difficult task to do in traditional data centre networks, in order to optimise energy consumption and SLA satisfaction, thus opening new research outlets [56]. Service Function Chaining (SFC) is an automated process to set up the chain of virtual network functions (VNFs), e.g., network address translation (NAT), firewalls, intrusion detection systems (IDS) in an NFV environment using instantiation of software-only services. Leveraging SDN together with NFV technologies allows for efficient and on-demand placement of service chains [47]. However, optimal service chain placement requires novel heuristics and resource management policies. The virtualized nature of VNFs also makes their orchestration and consolidation easier and dynamic deployment of network services possible [147, 179], calling for novel algorithms that can exploit these capabilities.

In addition, it is foreseeable that the ongoing interest for ML, deep learning, and AI applications will help in dealing with the complexity, heterogeneity, and scale, in addition to spawn novel research in established data centre resource management problems such as VM provisioning, consolidation, and load balancing. It is however important to recognise that potential loss of control and determinism may arise by adopting these techniques. Research in explainable AI may provide a suitable direction for novel research to facilitate the adoption of AI methods in Cloud management solutions within the industry [69].

For example, in scientific workflows the focus so far has been on efficiently managing the execution of platform-agnostic scientific applications. As the amount of data processed increases

and extreme-scale workflows begin to emerge, it is important to consider key concerns such as fault tolerance, performance modelling, efficient data management, and efficient resource usage. For this purpose, Big Data analytics will become a crucial tool [63]. For instance, monitoring and analysing resource consumption data may enable workflow management systems to detect performance anomalies and potentially predict failures, leveraging technologies such as serverless computing to manage the execution of complex workflows that are reusable and can be shared across multiple stakeholders. Although today there exist the technical possibility to define solutions of this kind, there is still a shortage of applications of serverless functions to HPC and scientific computing use cases, calling for further research in this space.

4.3 Reliability

One of the most challenging areas in Cloud computing systems is reliability as it has a great impact on the QoS as well as on the long term reputation of the service providers. Currently, all the Cloud services are provided based on the cost and performance of the services. The key challenge faced by Cloud service providers is how to deliver a competitive service that meets end users' expectations for performance, reliability, and QoS in the face of various types of independent as well as temporal and spatial correlated failures. So the future of research in this area will be focused on innovative Cloud services that provide reliability and resilience with assured service performance; which is called Reliability as a Service (RaaS). The main challenge is to develop a hierarchical and service-oriented cloud service reliability model based on advanced mathematical and statistical models [178]. This requires new modules to be included in the existing Cloud systems such as failure model and workload model to be adapted for resource provisioning policies and provide flexible reliability services to a wide range of applications.

One of the future directions in RaaS will be using deep and machine learning for failure prediction. This will be based on failure characterisation and development of a model from massive amount of failure datasets. Having a comprehensive failure prediction model will lead to a failure-aware resource provisioning that can guarantee the level of reliability and performance for the user's applications. This concept can be extended as another research direction for the Fog computing where there are several components on the edge. While fault-tolerant techniques such as replication could be a solution in this case, more efficient and intelligent approaches will be required to improve the reliability of new type of applications such as IoT applications. This needs to be incorporated with the power efficiency of such systems and solving this trade off will be a complex research challenge to tackle [162].

Another research direction in reliability will be about Cloud storage systems that are now mature enough to handle Big Data applications. However, failures are inevitable in Cloud storage systems as they are composed of large scale hardware components. Improving fault tolerance in Cloud storage systems for Big Data applications is a significant challenge. Replication and Erasure coding are the most important data reliability techniques employed in Cloud storage systems [166]. Both techniques have their own trade-offs in various parameters such as durability, availability, storage overhead, network bandwidth and traffic, energy consumption and recovery performance. Future research should include the challenges involved in employing both techniques in Cloud storage systems for Big Data applications with respect to the aforementioned parameters [166]. This hybrid technique applies proactive dynamic data replication of erasure coded data based on node failure prediction, which significantly reduces network traffic and improves the performance of Big Data applications with less storage overhead. So, the main research challenge would be solving a multivariable optimisation problem to take into account several metrics to meet users and providers requirements.

4.4 Sustainability

Sustainability of ICT systems is emerging as a major consideration [89] due to the energy consumption of ICT systems. Of course, sustainability also covers issues regarding the pollution and decontamination of the manufacturing and decommissioning of computer and network equipment, but this aspect is not covered in the present paper.

In response to the concern for sustainability, viewed primarily through the lens of energy consumption and energy awareness, increasingly large CDCs are being established, with up to 1000 MW of potential power consumption, in or close to areas where there are plentiful sources of renewable energy [27], such as hydro-electricity in northern Norway, and where natural cooling can be available as in areas close to the Arctic Circle. This actually requires new and innovative system architectures that can distribute data centres and Cloud computing, geographically. To address this, algorithms have been proposed, which rely on geographically distributed data coordination, resource provisioning and energy-aware and carbon footprint-aware provisioning in data centres [70, 111, 141]. In addition, geographical load balancing can provide an effective approach for optimising both performance and energy usage. With careful pricing, electricity providers can motivate Cloud service providers to "follow the renewables" and serve requests through CDCs located in areas where green energy is available [151]. On the other hand, the smart grid focuses on controlling the flow of energy in the electric grid with the help of computer systems and networks, and there seems to be little if any work on the energy consumption by the ICT components in the smart grid, perhaps because the amount would be small as compared to the overall energy consumption of a country or region. Interestingly enough, there has been recent work on dynamically coupling the flow of energy to computing and communication resources, and the flow of energy to the components of such computer/communication systems [91] in order to satisfy QoS and SLAs for jobs while minimising the energy consumption, but much more work will be needed.

However, placing data centres far away from most of the end users places a further burden on the energy consumption and QoS of the networks that connect the end users to the CDCs. Indeed, it is important to note that moving CDCs away from users will increase the energy consumed in networks, so that some remote solutions which are based on renewable energy may substantially increase the energy consumption of networks that are powered through conventional electrical supplies. Another challenge relates to the very short end-to-end delay that certain operations, such as financial transactions, require; thus data centres for financial services often need to be located in proximity to the actual human users and financial organisations (such as banks) that are designing, maintaining and modifying the financial decision making algorithms, as well as to the commodity trading data bases whose state must accurately reflect current prices, since users need to buy and sell stock or other commodities at up-to-date prices that may automatically change within less than a second. Another factor is the proprietary nature of the data that is being used, and the legal and security requirements that can often only be verified and complied within national boundaries or within the EU. Thus if the data remains local, the CDCs that process it also have to be local. Thus in many cases, the Cloud cannot rely on renewable energy to operate effectively simply because renewal energy is not available locally and because some renewable energy sources (e.g. wind and photovoltaic) tend to be intermittent. At the other end, the power needs of CDCs and the Cloud are also growing due to the ever-increasing amount of data that need to be stored and processed. Thus running the Cloud and CDCs in an energy efficient manner remains a major priority.

Unfortunately, high performance and more data processing has always gone hand-in-hand with greater energy consumption. Thus QoS, SLAs, and energy consumption have to be considered simultaneously and need to be managed online [92]. Since all the fast-changing online behaviours cannot be predicted in advance or modelled in a complete manner, adaptive self-aware techniques

are needed to face this challenge [223]. Some progress has been recently made in this direction [226] but further work will be needed. The actual algorithms that may be used will include machine learning techniques such as those described in Yin et al [236], which exploits constant online measurement of system parameters that can lead to online decision making that will optimise sustainability while respecting QoS considerations and SLAs.

The Fog can also substantially increase energy consumption because of the greater difficulty of efficient energy management for smaller and highly diverse systems [90, 93]. At the same time, the reduced access distance and network size from the end users to the Fog servers can create energy savings in networks. Therefore, the interesting trade-off between the increased energy consumption from many disparate and distribute Fog servers, and the reduced network energy consumption when the Fog servers are installed in close proximity to the end user, requires much further work [94]. Such research should include the improvements in network QoS that may be experienced by end users, when they access locally distributed Fog servers and their traffic traverses a smaller number of network nodes. There have been attempts to conduct experimental research in this direction with the help of machine learning based techniques [224].

Some approaches for improving sustainability and reducing energy consumption in the Cloud, primarily focus on the VM consolidation for minimising the energy consumption of the servers, which has been shown to be quite effective [25], while the Cloud cannot be accessed without the help of networks. However, reducing energy consumption in networks is also a complex problem [81, 96]. Saving energy for networking elements often disturbs other aspects such as reliability, scalability, and performance of the network [98]. Proposals have been made and tested regarding the design of smart energy-aware routing algorithms [95], but this area in general has received less attention compared to energy consumption and power efficiency of computing elements. With the advent of SDN, the global network awareness and centralised decision-making offered by SDN may provide a better opportunity for creating sustainable networks for Clouds [80]. This is perhaps one of the areas that will draw substantially more research efforts and innovation in the next decade.

4.5 Heterogeneity

Heterogeneity on the Cloud was introduced in the last decade, but awaits widespread adoption. As highlighted in Section 2.5, there are currently at least two significant gaps that hinder heterogeneity from being fully exploited on the Cloud. The first gap is between unified management platforms and heterogeneity. Existing research that targets resource and workload management in heterogeneous Cloud environments is fragmented. This translates into the lack of availability of a unified environment for efficiently exploiting VM level, vendor level and hardware architecture level heterogeneity while executing Cloud applications. The manifesto therefore proposes for the next decade an umbrella platform that accounts for heterogeneity at all three levels. This can be achieved by integrating a portfolio of workload and resource management techniques from which optimal strategies are selected based on the requirement of an application. For this, heterogeneous memory management will be required. Current solutions for memory management rely mainly on hypervisors, which limits the benefits from heterogeneity. Alternate solutions recently proposed rely on making guest operating systems heterogeneity-aware [138].

The second gap is between abstraction and heterogeneity. Current programming models for using hardware accelerators require accelerator specific languages and low level programming efforts. Moreover, these models are conducive for developing scientific applications. This restricts the wider adoption of heterogeneity for service oriented and user-driven applications on the Cloud. One meaningful direction to pursue will be to initiate a community-wide effort for developing an open-source high-level programming language that can satisfy core Cloud principles, such as abstraction and elasticity, which are suited for modern and innovative Cloud applications in a

heterogeneous environment. This will also be a useful tool as the Fog ecosystem emerges and applications migrate to incorporate both Cloud and Fog resources.

Recent research in this area has highlighted the limitation of current programming languages, such as OpenCL [43]. The interaction between CPUs and the hardware accelerator need to be explicitly programmed, which limits the automatic transformation of source code in efficient ways. To this end, fine-grained task partitioning needs to be automated for general purpose applications. Additionally, the automated conversion from coarse-grained to fine-grained task partitioning is required. In the context of OpenCL programming, there is limited performance portability, which is to be addressed. However, currently available high-level programming languages, such as TANGRAM [44] provide performance portability across different accelerators, but need to incorporate performance models and adaptive runtimes for finding optimal strategies for interaction between the CPU and the hardware accelerator.

Although the Cloud as a utility is a more recent offering, a number of the underlying technologies for supporting different levels of heterogeneity (memory, processors etc) in the Cloud came into inception a few decades ago. For example, the Multiplexed Information and Computing Service (Multics) offered single-level memory, which was the foundation of virtual memory for heterogeneous systems. Similarly, IBM developed CP-67, which was one of the first attempts in virtualizing mainframe operating systems to implement time-sharing. Later on VMWare used this technology for virtualizing x86 servers. The earlier technology was able to even provide I/O virtualization, and meaningful ways of addressing some of the challenges raised by modern heterogeneity may find inspiration in earlier technologies when the Cloud was not known.

Recently there is also a significant discussion about disaggregated data centres. Traditionally data centres are built using servers and racks with each server contributing the resources such as CPU, memory and storage, required for the computational tasks. With the disaggregated data centre each of these resources is built as a stand-alone resource "*blade*", where these blades are interconnected through a high-speed network fabric. The trend has come into existence as there is significant gap in the pace at which each of these resource technologies individually advanced. Even though most prototypes are proprietary and in their early stages of development, a successful deployment at the data centre level would have significant impact on the way the traditional IaaS are provided. However, this needs significant development in the network fabric as well [85].

4.6 Interconnected Clouds

As the grid computing and web service histories have shown, interoperability and portability across Cloud systems is a highly complicated area and it is clear at this time that pure standardisation is not sufficient to address this problem. The use of application containers and configuration management tools for portability, and the use of software adapters and libraries for interoperability are widely used as practical methods for achieving interoperation across Cloud services and products. However, there are a number of challenges [37], and thus potential research directions, that have been around since the early days of Cloud computing and, due to their complexity, have not been satisfactorily addressed so far.

One of such challenges is how to promote Cloud interconnection without forcing the adoption of the minimum common set of functionalities among services: if users want, they should be able to integrate complex functionalities even if they are offered only by one provider. Other research directions include how to enable Cloud interoperation middleware that can mimic complex services offered by one provider by composing simple services offered by one or more providers - so that the choice about the complex service or the composition of simpler services were solely dependent on the user constraints - cost, response time, data sovereignty, etc.

The above raises another important future research direction: how to enable middleware operating at the user-level (InterCloud and hybrid Clouds) to identify candidate services for a composition without support from Cloud providers? Given that providers have economic motivation to try to retain all the functionalities offered to their customers (i.e., they do not have motivation to facilitate that only some of the services in a composition are their own), one cannot expect that an approach that requires Cloud providers cooperation might succeed.

Therefore, the middleware enabling composition of services has to solve challenges in its two interfaces: in the interface with Cloud users, it needs to seamlessly deliver the service, in a level where how the functionality is delivered is not relevant for users: it could be obtained in all from a single provider (perhaps invoking a SaaS able to provide the functionality) or it could be obtained by composing different services from different providers. In the provider interface, it enables such more complex functions to be obtained, regardless of particular collaboration from providers: provided that an API exists, the middleware would be in charge of understanding what information/service the API can provide (and how to access such service) and thus decide by itself if it has all the required input necessary to access the API, and even the output is sufficient to enable the composition. This discussion makes clear the complexity of such middleware and the difficulty of the questions that need to be addressed to enable such vision.

Nevertheless, ubiquitously interconnected Clouds (achieved via Cloud Federation) can truly be achieved only when Cloud vendors are convinced that the Cloud interoperability adoption brings them financial and economic benefits. This requires novel approaches for billing and accounting, novel interconnected Cloud suitable pricing methods, along with formation of InterCloud marketplaces [209].

Finally, the emergence of SDNs and the capability to shape and optimise network traffic has the potential to influence research in Cloud interoperation. Google reports that one of the first uses of SDNs in the company was for optimisation of wide-area network traffic connecting their data centres [211]. In the same direction, investigation is needed on the feasibility and benefits of SDN and NFV to address some of the challenges above. For example, SDN and NFV can enable better security and QoS for services built as compositions of services from multiple providers (or from geographically distributed services from the same provider) by enforcing prioritization of service traffic across providers/data centres and specific security requirements [121].

4.7 Empowering Resource-Constrained Devices

Regarding future directions for empowering resource-constrained devices, in the mobile Cloud domain, we already have identified that, while task delegation is a reality, code offloading still has adaptability issues. It is also observed that, *"as the device capabilities are increasing, the applications that can benefit from the code offloading are becoming limited"* [201]. This is evident, as the capabilities of smartphones are increasing, to match or benefit from offloading, the applications are to be offloaded to Cloud instances with much higher capacity. This incurs higher cost per offloading. To address this, the future research in this domain should focus at better models for multi-tenancy in Mobile Cloud applications, to share the costs among multiple mobile users. The problem further gets complex due to the heterogeneity of both the mobile devices and Cloud resources.

We also foresee the need for incentive mechanisms for heterogeneous mobile Cloud offloading to encourage mobile users to participate and get appropriate rewards in return. This should encourage in adapting the mobile Cloud pattern to the social networking domain as well, in designing ideal scenarios. In addition, the scope and benefits offered by the emerging technologies such as serverless computing, CaaS and Fog computing, to the mobile Cloud domain, are not yet fully explored.

The incentive mechanisms are also relevant for the IoT and Fog domains. Recently there is significant discussion about the establishment of Fog closer to the *things*, by infrastructure offered

by independent Fog providers [42]. These architectures follow the consumer-as-provider (CaP) model. A relevant CaP example in the Cloud computing domain is the MQL5 Cloud Network [1], which utilises consumer's devices and desktops for performing various distributed computing tasks. Adaptation of such Peer-to-Peer (P2P) and CaP models would require ideal incentive mechanisms. Further discussion about the economic models for such Micro Data centres is provided in Section 4.9.

The container technology also brings several opportunities to this challenge. With the rise of Fog and Edge computing, it can be predicted that the container technology, as a kind of lightweight running environment and convenient packing tools for applications, will be widely deployed in edge servers. For example, the customised containers, such as Cloud Android Container [231], aimed at Edge computing and offloading features will be more and more popular. They provide efficient server runtime and inspire innovative applications in IoT, AI, and other promising fields.

Edge analytics in domains such as real-time streaming data analytics would be another interesting research direction for the resource constrained devices. The things in IoT primarily deal with sensor data and the Cloud-centric IoT (CIoT) model extracts this data and pushes it to the Cloud for processing. Primarily, Fog/Edge computing came to existence in order to reduce the network latencies in this model. In edge analytics, the sensor data will be processed across the complete hierarchy of Fog topology, i.e. at the edge devices, intermediate Fog nodes and Cloud. The intermediary processing tasks include filtering, consolidation, error detection etc. Frameworks that support edge analytics (e.g. Apache Edgent [10]) should be studied considering both the QoS and QoE (Quality of Experience) aspects. Preliminary solutions related to scheduling and placement of the edge analytics tasks and applications across the Fog topology are already appearing in the literature [155, 198]. Further research is required to deal with cost-effective multi-layer Fog deployment for multi-stage data analytics and dataflow applications.

4.8 Security and Privacy

Security and privacy issues are among the biggest concerns in adopting Cloud technologies. In particular, security and privacy issues are related to various technologies including, networks (Section 4.12), databases, virtualization, resource scheduling (Section 4.2), and so on. Possible solutions must be designed according to the specific trust assumptions at the basis of the considered scenario (e.g., a Cloud provider can be assumed completely untrusted/malicious, or it could be assumed trustworthy). In the following, we provide a brief description of future research directions in the security and privacy area, mainly focusing on problems related to the management of (sensitive) data.

Regarding the protection of data in the Cloud, we distinguish between two main scenarios of future research: 1) a simple scenario where the main problem is to guarantee the protection of data in storage as well as the ability to efficiently access and operate on them; 2) a scenario where data must be shared and accessed by multiple users and with the possible presence of multiple providers for better functionality and security. In the simple scenario, when data are protected with client-side encryption, there is the strong need for scalable and well-performing techniques that, while not affecting service functionality, can: 1) be easily integrated with current Cloud technology; 2) avoid possible information leakage caused by the solutions (e.g., indexes) adopted for selectively retrieving data or by the encryption supporting queries [169]; 3) support a rich variety of queries. Other challenges are related to the design of solutions completely departing from encryption and based on the splitting of data among multiple providers to guarantee generic confidentiality and access/visibility constraints possibly defined by the users. Considering the data integrity problem, an interesting research direction consists in designing solutions proving data integrity when data are distributed and stored on multiple independent Cloud providers. In the scenario with multiple users and possible multiple providers, a first issue to address is the design of

solutions for selectively sharing data that support: 1) write privileges as well as multiple writers; 2) the efficient enforcement of policies updates in distributed storage systems characterised by multiple and independent Cloud providers; 3) the selective sharing of information among parties involved in distributed computations, thus also taking advantage of the availability of cheaper (but not completely trusted) Cloud providers. The execution of distributed computations also requires the investigation of issues related to query privacy (which deals with the problem of protecting accesses to data) and computation integrity. Existing solutions for query privacy are difficult to apply in real-world scenarios for their computational complexity or for the limited kinds of queries supported. Interesting open issues are therefore the development of scalable and efficient techniques: i) supporting concurrent accesses by different users; and ii) ensuring no improper leakage on user activity and applicability in real database contexts. With respect to computation integrity, existing solutions are limited in their applicability, the integrity guarantees offered, and the kinds of supported queries. There is then the need to design a generic framework for evaluating the integrity guarantees provided according to the cost that a user is willing to pay to have such guarantees and that support different kinds of queries/computations. In presence of multiple Cloud providers offering similar services, it is critical for users to select the provider that better fits their need. Existing solutions supporting users in this selection process consider only limited user-based requirements (e.g., cost and performance requirements only) or pre-defined indicators. An interesting challenge is therefore the definition of a comprehensive framework that allows users both to express different requirements and preferences for the Cloud provider selection, and to verify that Cloud providers offer services fully compliant with the signed contract.

While emerging scenarios such as Fog Computing (Section 3.3) and Big Data (Section 3.2) have brought enormous benefits, as a side effect there is a tremendous exposure of private and sensitive information to privacy breaches. The lack of central controls in Fog-based scenarios may raise privacy and trust issues. Also, Fog computing assumes the presence of trusted nodes together with malicious ones. This requires adapting the earlier research of secure routing, redundant routing and trust topologies performed in the P2P context, to this novel setting [86]. While Cloud security research can rely on the idea that all data could be dumped into a data lake and analysed (in near real time) to spot security and privacy problems, this may no longer be possible when devices are not always connected and there are too many of them to make it financially viable to dump all the events into a central location. This Fog-induced fragmentation of information combined with encryption will foster a new wave of Cloud security research. Also the explosion of data and their variety (i.e., structured, unstructured, and semi-structured formats) make the definition and enforcement of scalable data protection solutions a challenging issue, especially considering the fact that the risk of inferring sensitive information significantly increases in Big Data. Other issues are related to the provenance and quality of Big Data. In fact, tracking Big Data provenance can be useful for: i) verifying whether data came from trusted sources and have been generated and used appropriately; and ii) evaluating the quality of the Big Data, which is particularly important in specific domains (e.g., healthcare). Blockchain technology can be helpful for addressing the data provenance challenge since it ensures that data in a blockchain are immutable, verifiable, and traceable. However, it also introduces novel privacy concerns since data (including personal data) in a blockchain cannot be changed or deleted.

At the infrastructure level, security and privacy issues that need to be further investigated include: the correct management of virtualization enabling multi-tenancy in the Cloud; the allocation and de-allocation of resources associated with virtual machines as well as the placement of virtual machine instances in the Cloud in accordance to security constraints imposed by users; and the identification of legitimate request to tackle issues such as Denial of Service (DoS) or other forms of cyber-attacks. These types of attacks are critical, as a coordinated attack on the Cloud services

can be wrongly inferred as legitimate traffic and the resources would be scaled up to handle them. This will result in both the incurred additional costs and waste in energy [197]. Cloud systems should be able to distinguish these attacks and decide either to drop the additional load or avoid excessive provisioning of resources. This requires extending the existing techniques of DDoS to also include exclusive characteristics of Cloud systems.

4.9 Economics of Cloud Computing

The economics of Cloud computing offers several interesting future research directions. As Cloud computing deployments based on VMs transition to the use of container-based deployments, there is increasing realisation that the lower overheads associated with container deployment can be used to support real-time workloads. Hence, serverless computing capability is now becoming commonplace with Google Cloud Functions, Amazon Lambda, Microsoft Azure Functions and IBM Bluemix OpenWhisk. In these approaches, no computing resources are actually charged for until a function is called. These functions are often simpler in scope and typically aimed at processing data stream-based workloads. The actual benefit of using serverless computing depends on the execution behaviour and types of workloads expected within an application. Eivy [71] outlines the factors that influence the economics of such function deployment, such as: (1) average vs. peak transaction rates; (2) scaling number of concurrent activity on the system, i.e. running multiple concurrent functions with increasing number of users; (3) benchmark execution of serverless functions on different backend hardware platforms, and the overall execution time required for your function.

Similarly, increasing usage of Fog and Edge computing capabilities alongside Cloud-based data centres offers significant research scope in Cloud economics. The combination of stable Cloud resources and volatile user edge resources can reduce the operating costs of Cloud services and infrastructures. However, we expect users to require some incentives to make their devices available at the edge. The availability of Fog and Edge resources provides the possibility for a number of additional business models and the inclusion of additional category of providers in the Cloud marketplace. We refer to the existence of such systems as Micro Data Centres (MDCs), which are placed between the more traditional data centre and user owned/provisioned resources. Business models include: (1) *Dynamic MDC discovery:* in this model, a user would dynamically be able to choose a MDC provider, according to the MDC availability profile, security credentials, or type. A service-based ecosystem with multiple such MDC providers may be realised, however this will not directly guarantee the fulfilment of the user objectives through integration of externally provisioned services. (2) *Pre-agreed MDC contracts:* in this model, detailed contracts adequately capture the circumstances and criteria that influence the performance of the MDC provisioned external services. A user's device would have these pre-agreed contracts or SLA with specific MDC operators, and would interact with them preferentially. This also reduces the potential risks incurred by the user. In performance-based contracts, an MDC would need to provide a minimum level of performance (e.g. availability) to the user which is reflected in the associated price. This could be achieved by interaction between MDCs being managed by the same operator, or by MDC outsourcing some of their tasks to a CDC; (3) *MDC federation:* in this model multiple MDC operators can collaborate to share workload within a particular area, and have preferred costs for exchange of such workload. This is equivalent to alliances established between airline operators to serve particular routes. To support such federation, security credentials between MDCs must be pre-agreed. This is equivalent to an extension of the pre-agreed MDC contracts business model, where MDCs across multiple coffee shop chains can be federated, offering greater potential choice for a user; (4) *MDC-Cloud data centre exchange:* in this model a user's device would contact a CDC in the first instance, which could then outsource computation to an MDC if it is unable to meet the required QoS targets (e.g. latency). A CDC could use any of the three approaches outlined above i.e. dynamic MDC discovery,

preferred MDCs, or choice of an MDC within a particular group. A CDC operator needs to consider whether outsourcing could still be profitable given the type of workload a user device is generating.

However, the unpredictable Cloud environment arising due to the use of Fog and Edge resources, and the dynamics of service provisioning in these environments, requires architects to embrace uncertainty. More specifically, architecting for the Cloud needs to strike a reasonable balance between dependable and efficient provision and their economics under uncertainties. In this context, the architecting process needs to incubate architecture design decisions that not only meet qualities such as performance, availability, reliability, security, compliance, among others, but also seek value through their provision. Research shall look at possible abstractions and formulations of the problem, where competitive and/or cooperative game design strategies can be explored to dynamically manage various players, including Cloud multitenants, service providers, resources etc. Future research should also explore Cloud architectures and market models that embrace uncertainties and provide continuous "win-win" resolutions (for providers, users and intermediaries) for value and dependability.

Similarly, migrating in-house IT systems (e.g. Microsoft Office 365 for managing email) and IT departments (e.g. systems management) to the Cloud also offers several research opportunities. What this migration means, longer term, for risk tolerance and business continuity remains unclear. Many argue that outsourcing of this kind gives companies access to greater levels of expertise (especially in cybersecurity, software updates, systems availability, etc.) compared to in-house management. However, issues around trust remain for many users – i.e. who can access their data and for what purpose. Recent regulations, such as the European GDPR and US CLOUD Act are aimed at addressing some of these concerns. The actual benefit of GDPR will probably not be known for a few years, as it comes into effect towards the end of May 2018.

The Edge analytics discussed in Section 4.7 also offers several research directions in this regard. Understanding what data should remain at or near user premises, and what should be migrated for analysis at a data centre remain important challenges. These also influence potential revenue models that could be developed taking account of a number of potential data storage/processing actors that would now exist from the data capture site to subsequent analysis within a CDC.

In addition, the Cloud Market place today is continuously expanding, with Cloud Harmony provider directory [53] reporting over 90 Cloud providers today. Although some providers dominate the market, there is still significant potential for new players to emerge, especially with recent emphasis on edge and serverless computing. Edge computing, in particular, opens up the potential market to telco operators who manage the mobile phone infrastructure. With increasing data volumes from emerging application areas such as autonomous vehicles and smart city sensing, such telco vendors are likely to form alliances with existing Cloud providers for supporting real time stream processing and edge analytics.

4.10 Application Development and Delivery

Agile, continuous, delivery paradigms often come at the expense of reduced reasoning at design-time on quality aspects such as SLA compliance, business alignment, and value-driven design, posing for example a risk of adopting the wrong architecture in the early design stages of a new Cloud application. These risks raise many research challenges on how to continuously monitor and iteratively evolve the design and quality of Cloud applications within the continuous delivery pipelines. The definition of supporting methods, high-level programming abstractions, tools and organisational processes to address these challenges is currently a limiting factor that requires further research. For example, it is important to extend existing software development and delivery methodologies with reusable abstractions for designing, orchestrating and managing IoT, Fog/Edge

computing, Big Data, and serverless computing technologies and platforms. Early efforts in these directions are already underway [39].

The trend towards using continuous delivery tools to automatically create, configure, and manage Cloud infrastructures (e.g., Chef, Ansible, Puppet, etc) through infrastructure-as-code is expected to continue and grow in the future years. However, there is still a fundamental shortage of software engineering methods specifically tailored to write, debug and evolve infrastructure-as-code. A challenge here is that infrastructure-as-code is often written in a combination of different programming and scripting languages, requiring greater generality than today in designing software quality engineering tools.

Another direction to extend existing approaches to Cloud application development and delivery is to define new architectural styles and Cloud-native design patterns to make Cloud application definition a process closer to human-thinking than today. The resulting software architectures and patterns need to take into account the runtime domain, and tolerate changes in contexts, situations, technologies, or service-level agreements leveraging the fact that, compared to traditional web services, emerging microservices and architectures offer simpler ways to automatically scale capacity, parallelism, and large-scale distribution, e.g., through microservices, serverless and FaaS.

Among the main challenges, the definition of novel architectures and patterns needs in particular to tackle Cloud application decomposition. The rapid growth of microservices and the fact that containers are becoming a de facto standard, raises the possibility to decompose an application in many more ways than in the past, with implications on its security, performance, reliability, and operational costs.

Further to this, with serverless computing and FaaS there will be the need for developing novel integration and control patterns to define services that combine traditional external services along with the serverless computing services. As an example, bridging in Edge computing the gap between cyber-physical systems (sensors, actuators, control layer) and the Cloud requires patterns to assist developers in building Cloudlets/swarmlets [149]. These are fragments of an application making local decisions and delegating tasks that cannot be solved locally to other Cloudlets/swarmlets in the Cloud [78], which are further discussed in Section 2.7. Developing effective Cloud design patterns also requires fundamental research on meta-controls for dynamic and seamless switching between these patterns at runtime, based on their value potentials and prospects. Such meta-controllers may rely on software models created by the application designers. Proposals in this direction include model-driven engines to facilitate reasoning, what-if analysis, monitoring feedback analysis, and for the correct enactment of adaptation decisions [23, 173].

Further research in patterns and architectures that combine multiple paradigms and technologies, will also require more work on formalisms to describe the user workload. Requirements in terms of performance, reliability, and security, need to be decomposed and propagated in architectures that combine emerging technologies (e.g., blockchain, SDN, Spark, Storm etc.) giving the ability not just to express execution requirements, but also to characterise the properties of the data processed by the application.

The trade-offs of orchestration of such integrated service mixes need to be investigated systematically considering the influence of the underpinning choice of Cloud resources (e.g., on-demand, reserved, spot, burstable) and the trade-off arising across multiple quality dimensions: (i) security (e.g., individual functions are easier to protect and verify than monoliths vs. greater attack surface with FaaS-based architectures); (ii) privacy (e.g., the benefits of model-based orchestration of access control vs greater data exposure in FaaS because of function calls and data flows); (iii) performance (e.g., the benefits of function-level autoscaling vs increased network traffic and latency experienced with FaaS); (iv) cost (e.g., FaaS cheaper to use per function invocation but can incur higher network charges than other architectural styles).

Research is also needed in programming models for adaptive elastic mobile decentralised distributed applications as needed by Fog/Edge computing, InterClouds, and the IoT. Separation of concerns will be important to address complexity of software development and delivery models. Functional application aspects should be specified, programmed, tested, and verified modularly. Program specifications may be probabilistic in nature, e.g., when analysing asynchronous data streams. Research is needed in specifying and verifying correctness of non-deterministic programs, which may result, e.g., from online machine learning algorithms. Non-functional aspects, e.g., fault tolerance, should be translucent: they can be completely left to the middleware, or applications should have declarative control over them, e.g., a policy favouring execution away from a mobile device in battery-challenged conditions [22]. *Translucent* programming models, languages, and Application Programming Interfaces (APIs) will be needed to enable tackling the complexity of application development while permitting control of application delivery to future-generation Clouds. One research direction to pursue will be the use of even finer-grained programming abstractions such as the actor model and associated middleware to dynamically reconfigure programs between edge resources and CDCs through transparent migration for users [125, 215].

4.11 Data Management

While Cloud IaaS and PaaS service for storage and data management focus on file, semi-structured and structured data independently, there is not much explicit focus on metadata management for datasets. Unlike structured data warehouses, the concept of "Data Lakes" encourages enterprises to put all their data into Cloud storage, such as HDFS, to allow knowledge to be mined from it. However, a lack of tracking metadata describing the source and provenance of the data makes it challenging to use them. Scientific repositories have over a decade of experience with managing large and diverse datasets along with the metadata that gives a context of use. Provenance that tracks the processing steps that have taken place to derive a data is also essential, for data quality, auditing and corporate governance. S3 offers some basic versioning capability, but metadata and provenance do not yet form a first-class entity in Cloud data platforms.

A key benefit of CDCs is the centralised collocation and management of data and compute at globally distributed data centres, offering economies of scale. The latency to access to data is however a challenge, along with bandwidth limitations across global networks. While Content Distribution Networks (CDN) such as AWS CloudFront cache data at regional level for web and video delivery, these are designed for slow-changing data and there is no such mechanism to write in data closer to the edge. Having Cloud data services at the Fog layer, which is a generalisation of CDN is essential. This is particularly a concern as IoT and 5G mobile networks become widespread.

In addition, Cloud storage has adapted to emerging security and privacy needs with support for HIPAA (Health Insurance Portability and Accountability Act of 1996) and other US CLOUD Act and EU GDPR regulations for data protection. However, enterprises that handle data that is proprietary and have sensitive trade secrets that can be compromised, if it is accessed by the Cloud provider, still remains a concern. While legal protections exist, there are no clear audit mechanisms to show that data has not been accessed by the Cloud provider themselves. Hybrid solutions where private data centres that are located near the public CDCs with dedicated high-bandwidth network allow users to manage sensitive data under their supervision while also leveraging the benefits of public Clouds [170].

Similarly, the interplay between hybrid models and SDN as well as joint optimisation of data flow placement, elasticity of Fog computing and flow routing can be better explored. Moreover, the computing capabilities of network devices can be leveraged to perform in-transit processing. The optimal placement of data processing applications and adaptation of dataflows, however, are

hard problems. This problem becomes even more challenging when considering the placement of stream processing tasks along with allocating bandwidth to meet latency requirements.

Furthermore, frameworks that provide high-level programming abstractions, such as Apache Bean, have been introduced in recent past to ease the development and deployment of Big Data applications that use hybrid models. Platform bindings have been provided to deploy applications developed using these abstractions on the infrastructure provided by commercial public Cloud providers such as Google Cloud Engine, Amazon Web Services, and open source solutions. Although such solutions are often restricted to a single cluster or data centre, efforts have been made to leverage resources from the edges of the Internet to perform distributed queries or to push frequently-performed analytics tasks to edge resources. This requires providing means to place data processing tasks in such environments while minimising the network resource usage and latency. In addition, efficient methods are to be investigated which manage resource elasticity in such scenarios. Moreover, high-level programming abstractions and bindings to platforms capable of deploying and managing resources under such highly distributed scenarios are desirable.

Lastly, there is a need to examine specialised data management services to support the trifecta of emerging disruptive technologies that will be hosted on Clouds: Internet of Things, Deep Learning, and Blockchain. As mentioned above, IoT will involve a heightened need to deal with streaming data, their efficient storage and the need to seamlessly incorporate data management on the edge seamlessly with management in the Cloud. Trust and provenance is particularly important when unmanaged edge devices play an active role.

The growing role of deep learning (see Section 3.7) will place importance on efficient management of trained models and their rapid loading and switching to support online and distributed analytics applications. Training of the models also requires access to large datasets, and this is particularly punitive for video and image datasets that are critical for applications like autonomous vehicles, and augmented reality. Novel data management techniques that offer compact storage and are also aware of the access patterns during training will be beneficial.

Lastly, Blockchain and distributed ledgers (see Section 3.6) can transform the way we manage and track data with increased assurance and provenance [45]. Financial companies (with crypto-currencies being just a popular manifestation) are at the forefront of using them for storing and tracking transactions, but these can also be extended to store other enterprise data in a secure manner with an implicit audit trail. Cloud-hosted distributed ledgers are already available as generic implementations (e.g. Ethereum and Hyperledger Fabric Blockchain platforms) but these are likely to be incorporated as integral part of Cloud data management. Another interesting area of research is in managing the ledger data itself in an efficient and scalable manner.

4.12 Networking

Global network view, programmability, and openness features of SDN provide a promising direction for application of SDN-based traffic engineering mechanisms within and across CDC networks. By using SDN within a data centre network, traffic engineering (TE) can be done much more efficiently and intelligently with dynamic flow scheduling and management based on current network utilisation and flow sizes [7]. Even though traffic engineering has been widely used in data networks, distinct features of SDN need a novel set of traffic engineering methods to utilise the available global view of the network and flow characteristics or patterns [6]. During the next decade we will also expect to see techniques targeting network performance requirements such as delay and bandwidth or even jitter guarantees to comply with QoS requirements of the Cloud user application and enforce committed SLAs.

SDN may also influence the security and privacy challenges in Cloud. In general, within the networking community, the overall perception is that SDN will help improve security and reliability

both within the network-layer and application-layer. As suggested by Kreutz et al [143], the capabilities brought by SDN may be used to introduce novel security services and address some of the on-going issues in Clouds. These include but are not limited to areas such as policy enforcement (for example, firewalling, access control, middleboxes), DoS attack detection and mitigation, monitoring infrastructures for fine-grained security examinations, and traffic anomaly detection.

Nevertheless, as a new technology, the paradigm shift brought by SDN brings along new threat vectors that may be used to target the network itself, services deployed on SDNs and the associated users. For instance, attackers may target the SDN controller as the single point of attack or the inter-SDN communications between the control and data plane - threats that did not exist in traditional networks. At the same time, the impact of existing threats may be magnified such as the range of capabilities available to an adversary who has compromised the network forwarding devices [190]. Hence, importing SDN to Clouds may impact the security of Cloud services in ways that have not been experienced or expected, which requires further research in this area.

The Cloud community has given significant priority to intra data centre networking, while efficient solutions for networking in interconnected environments are also highly demanded. Recent advances in SDN technology are expected to simplify intra data centre networking by making networks programmable and reduce both capital and operational expenditure for Cloud providers. However, the effectiveness of current approaches for interconnected Cloud environments and how SDN is used over public communication channels need further investigation.

One of the areas of networking that requires more attention is the management and orchestration of NFV environments. SFC is also a hot topic attaining a significant amount of attention by the community. So far, little attention has been paid to virtual network function (VNF) placement and consolidation while meeting the QoS requirements of the applications is highly desirable. Auto-scaling of VNFs within the service chains also requires in-depth attention. VNFs providing networking functions for the applications are subject to performance variation due to different factors such as the load of the service or overloaded underlying hosts. Therefore, development of auto-scaling mechanisms that monitor the performance of VNF instances and adaptively add or remove VNF instances to satisfy the SLA requirements of the applications is of paramount importance. Traffic engineering combined with migration and placement of VNFs provide a promising direction for the minimisation of network communication cost. Moreover, in auto-scaling techniques, the focus is often on auto-scaling of a single network service (e.g., firewall), while in practice auto-scaling of VNFs must be performed in accordance with service chains.

Recent advances in AI, ML, and Big Data analytics have great potential to address networking challenges of Cloud computing and automation of the next-generation networks in Clouds. The potential of these approaches along with centralised network visibility and readily accessible network information (e.g., network topology and traffic statistics) that SDN brings into picture, open up new opportunities to use ML and AI in networking. Even though it is still unclear how these can be incorporated into networking projects, we expect to see this as one of the exotic research areas in the following decade.

The emergence of IoT connecting billions of devices all generating data will place major demands on network infrastructure. 5G wireless and its bandwidth increase will also force significant expansion in network capacity with explosion in the number of mobile devices. Even though a key strategy in addressing latency and lower network resource usage is Edge/Fog computing, Edge/Fog computing itself is not enough to address all the networking demand. To meet the needs of this transition, new products and technologies expanding bandwidth, or the carrying capacity, of networks are required along with advances in faster broadband technologies and optical networking. Moreover, in both Edge and Fog computing, the integration of 5G so far has been discussed within a very narrow scope. Although 5G network resource management and resource

discovery in Edge/Fog computing have been investigated, many other challenging issues such as topology-aware application placement, dynamic fault detection, and network slicing management in this area are still unexplored.

4.13 Usability

There are several opportunities to enhance usability in Cloud environments. For instance, it is still hard for users to know how much they will spend renting resources due to workload/resource fluctuations or characteristics. Tools to have better estimations would definitely improve user experience and satisfaction. Due to recent demands from Big Data community, new visualization technologies could be further explored on the different layers of Cloud environment to better understand infrastructure and application behaviour and highlight insights to end users. Easier API management methodologies, tools, and standards are also necessary to handle users with different levels of expertise and interests. User experience when handling data-intensive applications also needs further studies considering their expected QoS.

In addition, users are still overloaded with resource and service types available to run their applications. Examples of resources and services are CPUs, GPUs, network, storage, operating system flavour, and all services available in the PaaS. Advisory systems to help these users would greatly enhance their experience consuming Cloud resources and services. Advisory systems to also recommend how users should use Cloud more efficiently would certainly be beneficial. Advices such as whether data should be transferred or visualized remotely, whether resources should be allocated or deleted, whether baremetal machines should replace virtual ones are examples of hints users could receive to make Cloud easier to use and more cost-effective.

The main difficulty in this area lies on evaluation. Traditionally, Cloud computing researchers and practitioners mostly perform quantitative experiments, whereas researchers working closer to users have deep knowledge on qualitative experiments. This second type of experiments depends on selecting groups of users with different profiles and investigating how they use technology. As Cloud has a very heterogeneous community of users with different needs and skills and work in different Cloud layers (IaaS, PaaS, and SaaS), such experiments are not trivial to be designed and executed at scale. Apart from understanding user behaviour, it is relevant to develop mechanisms to facilitate or automatically reconfigure Cloud technologies to adapt to the user needs and preferences, and not assume all users have the same needs or have the same level of skills.

4.14 Discussion

As can be observed from the emerging trends and proposed future research directions (summarised in the outer ring of Figure 3), there will be significant developments across all the service models (IaaS, PaaS and IaaS) of Cloud computing.

In the IaaS there is scope for heterogeneous hardware such as CPUs and accelerators (e.g. GPUs and TPUs) and special purpose Clouds for specific applications (e.g. HPC and deep learning). The future generation Clouds should also be ready to embrace the non-traditional architectures, such as neuromorphic, quantum computing, adiabatic, nanocomputing etc. Moreover, emerging trends such as containerisation, SDN and Fog/Edge computing are going to expand the research scope of IaaS by leaps and bounds. Solutions for addressing sustainability of CDC through utilisation of renewable energy and IoT-enabled cooling systems are also discussed. There is also scope for emerging trends in IaaS, such as disaggregated data centres where resources required for the computational tasks such as CPU, memory and storage, will be built as stand-alone resource blades, which will allow faster and ideal resource provisioning to satisfy different QoS requirements of Cloud based applications. The future research directions proposed for addressing the scalability,

Fig. 3. Future research directions in the Cloud computing horizon

resource management and scheduling, heterogeneity, interconnected Clouds and networking challenges, should enable realising such comprehensive IaaS offered by the Clouds.

Similarly, PaaS should see significant advancements through future research directions in resource management and scheduling. The need for programming abstractions, models, languages and systems supporting scalable elastic computing and seamless use of heterogeneous resources are proposed leading to energy-efficiency, minimised application engineering cost, better portability and guaranteed level of reliability and performance. It is also foreseeable that the ongoing interest for ML, deep learning, and AI applications will help in dealing with the complexity, heterogeneity, scale and load balancing applications developed through PaaS. Serverless computing is an emerging trend in PaaS, which is a promising area to be explored with significant practical and economic impact. Interesting future directions are proposed such as function-level QoS management and

economics for serverless computing. In addition, future research directions for data management and analytics are also discussed in detail along with security, leading to interesting applications with platform support such as edge analytics for real-time stream data processing, from the IoT and smart cities domains.

SaaS should mainly see advances from the application development and delivery, and usability of Cloud services. Translucent programming models, languages, and APIs will be needed to enable tackling the complexity of application development while permitting control of application delivery to future-generation Clouds. A variety of agile delivery tools and Cloud standards (e.g., TOSCA) are increasingly being adopted during Cloud application development. The future research should focus at how to continuously monitor and iteratively evolve the design and quality of Cloud applications. It is also suggested to extend DevOps methods and define novel programming abstractions to include within existing software development and delivery methodologies, a support for IoT, Edge computing, Big Data, and serverless computing. Focus should also be at developing effective Cloud design patterns and development of formalisms to describe the workloads and workflows that the application processes, and their requirements in terms of performance, reliability, and security are strongly encouraged. It is also interesting to see that even though the technologies have matured, certain domains such as mobile Cloud, still have adaptability issues. Novel incentive mechanisms are required for mobile Cloud adaptability as well as for designing Fog architectures.

Future research should thus explore Cloud architectures and market models that embrace uncertainties and provide continuous "win-win" resolutions, for all the participants including providers, users and intermediaries, both from the Return On Investment (ROI) and satisfying SLA perspectives.

5 SUMMARY AND CONCLUSIONS

The Cloud computing paradigm has revolutionised the computer science horizon during the past decade and enabled emergence of computing as the fifth utility. It has emerged as the backbone of modern economy by offering subscription-based services anytime, anywhere following a pay-as-you-go model. Thus, Cloud computing has enabled new businesses to be established in a shorter amount of time, has facilitated the expansion of enterprises across the globe, has accelerated the pace of scientific progress, and has led to the creation of various models of computation for pervasive and ubiquitous applications, among other benefits.

However, the next decade will bring about significant new requirements, from large-scale heterogeneous IoT and sensor networks producing very large data streams to store, manage, and analyse, to energy- and cost-aware personalised computing services that must adapt to a plethora of hardware devices while optimising for multiple criteria including application-level QoS constraints and economic restrictions. These requirements will be posing several new challenges in Cloud computing and will be creating the need for new approaches and research strategies, and force us to re-evaluate the models that were already developed to address the issues such as scalability, resource provisioning, and security.

This comprehensive manifesto brought the advancements together and proposed the challenges still to be addressed in realising the future generation Cloud computing. In the process, the manifesto identified the current major challenges in Cloud computing domain and summarised the state-of-the-art along with the limitations. The manifesto also discussed the emerging trends and impact areas that further drive these Cloud computing challenges. Having identified these open issues, the manifesto then offered comprehensive future research directions in the Cloud computing horizon for the next decade. The discussed research directions show a promising and exciting future for the Cloud computing field both technically and economically, and the manifesto calls the community for action in addressing them.

ACKNOWLEDGMENTS

We thank anonymous reviewers, Sartaj Sahni (Editor-in-Chief) and Antonio Corradi (Associate Editor) for their constructive suggestions and guidance on improving the content and quality of this paper. We also thank Adam Wierman (California Institute of Technology), Shigeru Imai (Rensselaer Polytechnic Institute) and Arash Shaghaghi (University of New South Wales, Sydney) for their comments and suggestions for improving the paper. Regarding funding, G. Casale has been supported by the Horizon 2020 project DICE (644869).

REFERENCES

[1] 2017. MQL5 Cloud Network. https://cloud.mql5.com/. (2017). [Last visited on 18th May 2018].
[2] 2017. UberCloud application containers. https://www.TheUberCloud.com/containers/. (2017). [Last visited on 18th May 2018].
[3] 2017. Unikernels - Rethinking Cloud Infrastructure. http://unikernel.org/. (2017). [Last visited on 18th May 2018].
[4] R. Agrawal, J. Kierman, R. Srikant, and Y. Xu. 2004. Order Preserving Encryption for Numeric Data. In *Proc. of SIGMOD 2004*. Paris, France.
[5] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. 2004. Order preserving encryption for numeric data. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*. ACM, 563–574.
[6] Ian F Akyildiz, Ahyoung Lee, Pu Wang, Min Luo, and Wu Chou. 2014. A roadmap for traffic engineering in SDN-OpenFlow networks. *Computer Networks* 71 (2014), 1–30.
[7] Mohammad Al-Fares, Sivasankar Radhakrishnan, Barath Raghavan, Nelson Huang, and Amin Vahdat. 2010. Hedera: Dynamic Flow Scheduling for Data Center Networks.. In *NSDI*, Vol. 10. 19–19.
[8] Amazon Web Services. 2015. *AWS Serverless Multi-Tier Architectures - Using Amazon API Gateway and AWS Lambda*. Technical Report. Amazon Web Services.
[9] Jonatha Anselmi, Danilo Ardagna, John Lui, Adam Wierman, Yunjian Xu, and Zichao Yang. 2017. The Economics of the Cloud. *ACM Trans. on Modeling and Performance Evaluation of Computing Systems (TOMPECS)* 2, 4 (2017), 18.
[10] Apache Software Foundation. 2018. Apache Edgent - A Community for Accelerating Analytics at the Edge. http://edgent.apache.org/. (2018). [Last visited on 18th May 2018].
[11] Arvind Arasu, Spyros Blanas, Ken Eguro, Raghav Kaushik, Donald Kossmann, Ravishankar Ramamurthy, and Ramarathnam Venkatesan. 2013. Orthogonal Security with Cipherbase. In *CIDR*.
[12] Danilo Ardagna, Giuliano Casale, Michele Ciavotta, Juan F Pérez, and Weikun Wang. 2014. Quality-of-service in cloud computing: modeling techniques and their applications. *Jour of Internet Services and Applications* 5, 1 (2014), 11.
[13] Sergei Arnautov, Bohdan Trach, Franz Gregor, Thomas Knauth, Andre Martin, Christian Priebe, Joshua Lind, Divya Muthukumaran, Dan O'Keeffe, Mark Stillwell, et al. 2016. SCONE: Secure Linux Containers with Intel SGX.. In *OSDI*. 689–703.
[14] Matt Asay. 2018. AWS won serverless Ű now all your software are kinda belong to them. https://www.theregister.co.uk/2018/05/11/lambda_means_game_over_for_serverless/. (May 2018). [Last visited on 18th May 2018].
[15] Siamak Azodolmolky, Philipp Wieder, and Ramin Yahyapour. 2013. Cloud computing networking: challenges and opportunities for innovations. *IEEE Communications Magazine* 51, 7 (2013), 54–62.
[16] Enrico Bacis, Sabrina De Capitani di Vimercati, Sara Foresti, Stefano Paraboschi, Marco Rosa, and Pierangela Samarati. 2016. Mix&Slice: Efficient access revocation in the cloud. In *ACM SIGSAC Conf. on Computer and Communications Security*. 217–228.
[17] Arshdeep Bahga and Vijay K Madisetti. 2016. Blockchain platform for industrial Internet of Things. *J. Softw. Eng. Appl* 9, 10 (2016), 533.
[18] Armin Balalaie, Abbas Heydarnoori, and Pooyan Jamshidi. 2016. Microservices architecture enables DevOps: migration to a cloud-native architecture. *IEEE Software* 33, 3 (2016), 42–52.
[19] Ioana Baldini, Paul Castro, Kerry Chang, Perry Cheng, Stephen Fink, Vatche Ishakian, Nick Mitchell, Vinod Muthusamy, Rodric Rabbah, Aleksander Slominski, et al. 2017. Serverless Computing: Current Trends and Open Problems. *arXiv preprint arXiv:1706.03178* (2017).
[20] Akindele A Bankole and Samuel A Ajila. 2013. Predicting cloud resource provisioning using machine learning techniques. In *Electrical and Computer Engineering (CCECE), 2013 26th Annual IEEE Canadian Conference on*. IEEE, 1–4.
[21] Len Bass, Ingo Weber, and Liming Zhu. 2015. *DevOps: A Software Architect's Perspective*. Addison-Wesley Professional.
[22] Anton Beloglazov, Jemal Abawajy, and Rajkumar Buyya. 2012. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future generation computer systems* 28, 5 (2012), 755–768.

[23] Alexander Bergmayr, Uwe Breitenbücher, Nicolas Ferry, Alessandro Rossini, Arnor Solberg, Manuel Wimmer, Gerti Kappel, and Frank Leymann. 2018. A Systematic Review of Cloud Modeling Languages. *ACM Comput. Surv.* 51, 1 (2018), 22:1–22:38. https://doi.org/10.1145/3150227

[24] Andreas Berl, Erol Gelenbe, Marco Di Girolamo, Giovanni Giuliani, Hermann De Meer, Minh Quan Dang, and Kostas Pentikousis. 2010. Energy-efficient cloud computing. *The computer journal* 53, 7 (2010), 1045–1051.

[25] A. Berl, E. Gelenbe, M. DiGirolamo, G. Giuliani, H. DeMeer, and M. Q. Dang. 2010. Energy-efficient Cloud computing. *Comput. J.* 53, 7 (2010), 1045–1051.

[26] David Bernstein, Erik Ludvigson, Krishna Sankar, Steve Diamond, and Monique Morrow. 2009. Blueprint for the intercloud-protocols and formats for cloud computing interoperability. In *Int. Conf. on Internet and Web Applications and Services (ICIW'09)*. IEEE, 328–336.

[27] Josep L Berral, Íñigo Goiri, Thu D Nguyen, Ricard Gavalda, Jordi Torres, and Ricardo Bianchini. 2014. Building green cloud services at low cost. In *IEEE 34th Int. Conf. on Distributed Computing Systems (ICDCS)*. IEEE, 449–460.

[28] Christian Bizer, Tom Heath, and Tim Berners-Lee. 2009. Linked data-the story so far. *Semantic services, interoperability and web applications: emerging concepts* (2009), 205–227.

[29] Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. 2012. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. ACM, 13–16.

[30] Nicolas Bonvin, Thanasis G Papaioannou, and Karl Aberer. 2011. Autonomic SLA-driven provisioning for cloud applications. In *IEEE/ACM int. symp. on cluster, cloud and grid computing*. IEEE Computer Society, 434–443.

[31] Z. Brakerski and V. Vaikuntanathan. 2011. Efficient Fully Homomorphic Encryption from (standard) LWE. In *Proc. of FOCS*. Palm Springs, CA, USA.

[32] Ross Brewer. 2014. Advanced persistent threats: minimising the damage. *Network Security* 2014, 4 (2014), 5–9.

[33] Rajkumar Buyya and Diana Barreto. 2015. Multi-cloud resource provisioning with aneka: A unified and integrated utilisation of microsoft azure and amazon EC2 instances. In *Computing and Network Communications (CoCoNet), 2015 International Conference on*. IEEE, 216–229.

[34] Rajkumar Buyya, Anton Beloglazov, and Jemal Abawajy. 2010. Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges. In *Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2010)*. CSREA Press.

[35] Rajkumar Buyya, Rodrigo N Calheiros, Jungmin Son, Amir Vahid Dastjerdi, and Young Yoon. 2014. Software-defined cloud computing: Architectural elements and open challenges. In *International Conference on Advances in Computing, Communications and Informatics (ICACCI 2014)*. IEEE, 1–12.

[36] Rajkumar Buyya, Saurabh Kumar Garg, and Rodrigo N Calheiros. 2011. SLA-oriented resource provisioning for cloud computing: Challenges, architecture, and solutions. In *Cloud and Service Computing (CSC), 2011 International Conference on*. IEEE, 1–10.

[37] Rajkumar Buyya, Rajiv Ranjan, and Rodrigo N Calheiros. 2010. Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services. In *Int. Conf. on Algorithms and Architectures for Parallel Processing*. Springer, 13–31.

[38] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. 2009. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation computer systems* 25, 6 (2009), 599–616.

[39] Giuliano Casale, Cristina Chesta, Peter Deussen, Elisabetta Di Nitto, Panagiotis Gouvas, Sotiris Koussouris, Vlado Stankovski, Andreas Symeonidis, Vlassis Vlassiou, Anastasios Zafeiropoulos, et al. 2016. Current and Future Challenges of Software Engineering for Services and Applications. *Procedia Computer Science* 97 (2016), 34–42.

[40] Emiliano Casalicchio and Luca Silvestri. 2013. Mechanisms for SLA provisioning in cloud-based service providers. *Computer Networks* 57, 3 (2013), 795–810.

[41] Israel Casas, Javid Taheri, Rajiv Ranjan, and Albert Y Zomaya. 2017. PSO-DS: a scheduling engine for scientific workflow managers. *The Journal of Supercomputing* 73, 9 (2017), 3924–3947.

[42] Chii Chang, Satish Narayana Srirama, and Rajkumar Buyya. 2017. Indie Fog: An Efficient Fog-Computing Infrastructure for the Internet of Things. *IEEE Computer* 50, 9 (2017), 92–98.

[43] Li-Wen Chang, Juan Gómez-Luna, Izzat El Hajj, Sitao Huang, Deming Chen, and Wen-mei Hwu. 2017. Collaborative Computing for Heterogeneous Integrated Systems. In *Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering (ICPE '17)*. 385–388.

[44] L. W. Chang, I. E. Hajj, C. Rodrigues, J. Gómez-Luna, and W. m. Hwu. 2016. Efficient kernel synthesis for performance portable programming. In *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. 1–13.

[45] Steve Cheng, Matthias Daub, Axel Domeyer, and Martin Lundqvist. 2017. Using blockchain to improve data management in the public sector. https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/using-blockchain-to-improve-data-management-in-the-public-sector. (2017). [Last visited on 18th May 2018].

[46] Margaret Chiosi, Don Clarke, Peter Willis, Andy Reid, James Feger, Michael Bugenhagen, Waqar Khan, Michael Fargano, Chunfeng Cui, Hui Deng, et al. 2012. Network functions virtualisation: An introduction, benefits, enablers, challenges and call for action. In *SDN and OpenFlow World Congress*. 22–24.
[47] Daewoong Cho, Javid Taheri, Albert Y Zomaya, and Pascal Bouvry. 2017. Real-Time Virtual Network Function (VNF) Migration toward Low Network Latency in Cloud Environments. In *Cloud Computing (CLOUD), 2017 IEEE 10th International Conference on*. IEEE, 798–801.
[48] Minsik Cho, Ulrich Finkler, Sameer Kumar, David S. Kung, Vaibhav Saxena, and Dheeraj Sreedhar. 2017. PowerAI DDL. *CoRR* abs/1708.02188 (2017). http://arxiv.org/abs/1708.02188
[49] Byung-Gon Chun, Sunghwan Ihm, Petros Maniatis, Mayur Naik, and Ashwin Patti. 2011. Clonecloud: elastic execution between mobile device and cloud. In *Proceedings of the sixth conference on Computer systems*. ACM, 301–314.
[50] Philip Church, Andrzej Goscinski, and Christophe Lefèvre. 2015. Exposing HPC and sequential applications as services through the development and deployment of a SaaS cloud. *Future Generation Computer Systems* 43 (2015), 24–37.
[51] Valentina Ciriani, Sabrina De Capitani Di Vimercati, Sara Foresti, Sushil Jajodia, Stefano Paraboschi, and Pierangela Samarati. 2010. Combining fragmentation and encryption to protect privacy in data storage. *ACM Transactions on Information and System Security (TISSEC)* 13, 3 (2010), 22.
[52] Cloud Standards Customer Council. 2017. *Cloud Customer Architecture for Blockchain*. Technical Report.
[53] CloudHarmony. 2018. CloudSquare - Provider Directory. https://cloudharmony.com/directory. (2018). [Last visited on 18th May 2018].
[54] Coupa Software. 2012. *Usability in Enterprise Cloud Applications*. Technical Report. Coupa Software.
[55] Steve Crago, Kyle Dunn, Patrick Eads, Lorin Hochstein, Dong-In Kang, Mikyung Kang, Devendra Modium, Karandeep Singh, Jinwoo Suh, and John Paul Walters. 2011. Heterogeneous cloud computing. In *Cluster Computing (CLUSTER), 2011 IEEE International Conference on*. IEEE, 378–385.
[56] Richard Cziva, Simon Jouët, David Stapleton, Fung Po Tso, and Dimitrios P Pezaros. 2016. SDN-based virtual machine management for cloud data centers. *IEEE Transactions on Network and Service Management* 13, 2 (2016), 212–225.
[57] Ernesto Damiani, SDCD Vimercati, Sushil Jajodia, Stefano Paraboschi, and Pierangela Samarati. 2003. Balancing confidentiality and efficiency in untrusted relational DBMSs. In *Proceedings of the 10th ACM conference on Computer and communications security*. ACM, 93–102.
[58] Amir Vahid Dastjerdi and Rajkumar Buyya. 2014. Compatibility-aware cloud service composition under fuzzy preferences of users. *IEEE Transactions on Cloud Computing* 2, 1 (2014), 1–13.
[59] Amir Vahid Dastjerdi and Rajkumar Buyya. 2016. Fog computing: Helping the Internet of Things realize its potential. *Computer* 49, 8 (2016), 112–116.
[60] Sabrina De Capitani di Vimercati, Sara Foresti, Sushil Jajodia, Stefano Paraboschi, and Pierangela Samarati. 2016. Efficient integrity checks for join queries in the cloud. *Journal of Computer Security* 24, 3 (2016), 347–378.
[61] S De Capitani di Vimercati, Giovanni Livraga, Vincenzo Piuri, Pierangela Samarati, and Gerson A Soares. 2016. Supporting application requirements in cloud-based iot information processing. In *International Conference on Internet of Things and Big Data (IoTBD 2016)*. Scitepress, 65–72.
[62] Jeffrey Dean. 2009. Large-scale distributed systems at google: Current systems and future directions. In *The 3rd ACM SIGOPS International Workshop on Large Scale Distributed Systems and Middleware (LADIS 2009) Tutorial*.
[63] Ewa Deelman, Christopher Carothers, Anirban Mandal, Brian Tierney, Jeffrey S Vetter, Ilya Baldin, Claris Castillo, Gideon Juve, et al. 2017. PANORAMA: an approach to performance modeling and diagnosis of extreme-scale workflows. *The International Journal of High Performance Computing Applications* 31, 1 (2017), 4–18.
[64] Travis Desell, Malik Magdon-Ismail, Boleslaw Szymanski, Carlos Varela, Heidi Newberg, and Nathan Cole. 2009. Robust asynchronous optimization for volunteer computing grids. In *e-Science, 2009. e-Science'09. Fifth IEEE International Conference on*. IEEE, 263–270.
[65] Sabrina De Capitani di Vimercati, Sara Foresti, Riccardo Moretti, Stefano Paraboschi, Gerardo Pelosi, and Pierangela Samarati. 2016. A dynamic tree-based data structure for access privacy in the cloud. In *Cloud Computing Technology and Science (CloudCom), 2016 IEEE International Conference on*. IEEE, 391–398.
[66] Angel Diaz. 2016. Three Ways That "Serverless" Computing Will Transform App Development In 2017. https://www.forbes.com/sites/ibm/2016/11/17/three-ways-that-serverless-computing-will-transform-app-development-in-2017/. (2016). [Last visited on 18th May 2018].
[67] Hoang T Dinh, Chonho Lee, Dusit Niyato, and Ping Wang. 2013. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing* 13, 18 (2013), 1587–1611.
[68] Docker Inc. 2018. Docker Swarm. https://docs.docker.com/swarm/. (2018). [Last visited on 18th May 2018].
[69] Derek Doran, Sarah Schulz, and Tarek R. Besold. 2017. What Does Explainable AI Really Mean? A New Conceptualization of Perspectives. *arXiv preprint, arXiv:1710.00794* (2017).

[70] Hancong Duan, Chao Chen, Geyong Min, and Yu Wu. 2017. Energy-aware Scheduling of Virtual Machines in Heterogeneous Cloud Computing Systems. *Future Generation Computer Systems* 74 (2017), 142 – 150.
[71] Adam Eivy. 2017. Be Wary of the Economics of "Serverless" Cloud Computing. *IEEE Cloud Computing* 4, 2 (2017), 6–12.
[72] Vincenzo Eramo, Emanuele Miucci, Mostafa Ammar, and Francesco Giacinto Lavacca. 2017. An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures. *IEEE/ACM Transactions on Networking* (2017).
[73] Dave Evans. 2011. The internet of things: How the next evolution of the internet is changing everything. *CISCO white paper* 1, 2011 (2011), 1–11.
[74] Chaudhry Muhammad Nadeem Faisal. 2011. Issues in Cloud Computing: Usability evaluation of Cloud based application. (2011).
[75] Funmilade Faniyi and Rami Bahsoon. 2016. A systematic review of service level management in the cloud. *ACM Computing Surveys (CSUR)* 48, 3 (2016), 43.
[76] W. Felter, A. Ferreira, R. Rajamony, and J. Rubio. 2015. An updated performance comparison of virtual machines and Linux containers. In *2015 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. 14pp. https://doi.org/10.1109/ISPASS.2015.7095802
[77] Huber Flores, Pan Hui, Sasu Tarkoma, Yong Li, Satish Srirama, and Rajkumar Buyya. 2015. Mobile code offloading: from concept to practice and beyond. *IEEE Communications Magazine* 53, 3 (2015), 80–88.
[78] Huber Flores and Satish Narayana Srirama. 2014. Mobile cloud middleware. *Journal of Systems and Software* 92 (2014), 82–94.
[79] Geoffrey C. Fox, Vatche Ishakian, Vinod Muthusamy, and Aleksander Slominski. 2017. Status of Serverless Computing and Function-as-a-Service(FaaS) in Industry and Research. *CoRR* abs/1708.08028 (2017). http://arxiv.org/abs/1708.08028
[80] Frederic Francois and Erol Gelenbe. 2016. Towards a Cognitive Routing Engine for Software Defined Networks. In *IEEE International Conference on Communications*. IEEE. https://doi.org/10.1109/ICC.2016.7511138
[81] F. Francois, N. Wang, K. Moessner, S. Georgoulas, and R. de Oliveira-Schmidt. 2014. Leveraging MPLS backup paths for distributed energy-aware traffic engineering. *IEEE Transactions on Network and Service Management* 11, 2 (2014), 235–249.
[82] Ivo Friedberg, Florian Skopik, Giuseppe Settanni, and Roman Fiedler. 2015. Combating advanced persistent threats: From network event correlation to incident detection. *Computers & Security* 48 (2015), 35–57.
[83] Edoardo Gaetani, Leonardo Aniello, Roberto Baldoni, Federico Lombardi, Andrea Margheri, and Vladimiro Sassone. 2017. Blockchain-Based Database to Ensure Data Integrity in Cloud Computing Environments.. In *ITASEC*. 146–155.
[84] Jim Gao and Ratnesh Jamidar. 2014. Machine learning applications for data center optimization. *Google White Paper* (2014).
[85] Peter Xiang Gao, Akshay Narayan, Sagar Karandikar, Joao Carreira, Sangjin Han, Rachit Agarwal, Sylvia Ratnasamy, and Scott Shenker. 2016. Network Requirements for Resource Disaggregation. In *OSDI*. 249–264.
[86] Pedro Garcia Lopez, Alberto Montresor, Dick Epema, Anwitaman Datta, Teruo Higashino, Adriana Iamnitchi, Marinho Barcellos, Pascal Felber, and Etienne Riviere. 2015. Edge-centric computing: Vision and challenges. *ACM SIGCOMM Computer Communication Review* 45, 5 (2015), 37–42.
[87] Saurabh Kumar Garg, Steve Versteeg, and Rajkumar Buyya. 2013. A framework for ranking of cloud computing services. *Future Generation Computer Systems* 29, 4 (2013), 1012–1023.
[88] Erol Gelenbe. 2014. Adaptive management of energy packets. In *Computer Software and Applications Conference Workshops (COMPSACW), 2014 IEEE 38th International*. IEEE, 1–6.
[89] Erol Gelenbe and Yves Caseau. 2015. The impact of information technology on energy consumption and carbon emissions. *Ubiquity* 2015, June (2015), 1.
[90] Erol Gelenbe and Elif Tugce Ceran. 2016. Energy packet networks with energy harvesting. *IEEE Access* 4 (2016), 1321–1331.
[91] Erol Gelenbe and E. T. Ceran. 2016. Energy packet networks with energy harvesting. *IEEE Access* 4 (2016), 1321–1331.
[92] Erol Gelenbe and Ricardo Lent. 2012. Optimising server energy consumption and response time. *Theoretical and Applied Informatics* 24, 4 (2012), 257–270.
[93] Erol Gelenbe and Ricardo Lent. 2013. EnergyŰqos trade-offs in mobile service selection. *Future Internet* 5, 2 (2013), 128–139.
[94] Erol Gelenbe, Ricardo Lent, and Markos Douratsos. [n. d.]. Choosing a local or remote cloud. In *Network Cloud Computing and Applications (NCCA), 2012 Second Symposium on* (2012). IEEE, 25–30.
[95] Erol Gelenbe and Toktam Mahmoodi. 2011. Energy-aware routing in the cognitive packet network. *Energy* 5 (2011), 7–12.
[96] Erol Gelenbe and Christina Morfopoulou. 2011. A framework for energy-aware routing in packet networks. *Comput. J.* 54, 6 (2011), 850–859.

[97] Erol Gelenbe and Christina Morfopoulou. 2012. Power savings in packet networks via optimised routing. *Mobile Networks and Applications* 17, 1 (2012), 152–159.
[98] Erol Gelenbe and Simone Silvestri. 2009. Reducing power consumption in wired networks. In *Computer and Information Sciences, 2009. ISCIS 2009. 24th International Symposium on.* IEEE, 292–297.
[99] C. Gentry. 2009. Fully Homomorphic Encryption using Ideal Lattices. In *Proc. of STOC*. Bethesda, MD, USA.
[100] C. Gentry, A. Sahai, and B. Waters. 2013. Homomorphic Encryption from Learning with Errors: Conceptually-simpler, Asymptotically-faster, Attribute-based. In *Proc. of CRYPTO*. Santa Barbara, CA, USA.
[101] Wolfgang Gentzsch and Burak Yenier. 2016. Novel Software Containers for Engineering and Scientific Simulations in the Cloud. *International Journal of Grid and High Performance Computing (IJGHPC)* 8, 1 (2016), 38–49.
[102] Ali Ghodsi, Matei Zaharia, Benjamin Hindman, Andy Konwinski, Scott Shenker, and Ion Stoica. 2011. Dominant Resource Fairness: Fair Allocation of Multiple Resource Types.. In *Nsdi*, Vol. 11. 24–24.
[103] Rahul Ghosh, Kishor S Trivedi, Vijay K Naik, and Dong Seong Kim. 2010. End-to-end performability analysis for infrastructure-as-a-service cloud: An interacting stochastic models approach. In *Dependable Computing (PRDC), 2010 IEEE 16th Pacific Rim International Symposium on.* IEEE, 125–132.
[104] Albert Greenberg, James R. Hamilton, Navendu Jain, Srikanth Kandula, Changhoon Kim, Parantap Lahiri, David A. Maltz, Parveen Patel, and Sudipta Sengupta. 2009. VL2: A Scalable and Flexible Data Center Network. *SIGCOMM Comput. Commun. Rev.* 39, 4 (Aug. 2009), 51–62. https://doi.org/10.1145/1594977.1592576
[105] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. 2013. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems* 29, 7 (2013), 1645–1660.
[106] Haryadi S Gunawi, Thanh Do, Joseph M Hellerstein, Ion Stoica, Dhruba Borthakur, and Jesse Robbins. 2011. Failure as a service (faas): A cloud service for large-scale, online failure drills. *University of California, Berkeley* 3 (2011).
[107] Chuanxiong Guo, Guohan Lu, Dan Li, Haitao Wu, Xuan Zhang, Yunfeng Shi, Chen Tian, Yongguang Zhang, and Songwu Lu. 2009. BCube: A High Performance, Server-centric Network Architecture for Modular Data Centers. *SIGCOMM Comput. Commun. Rev.* 39, 4 (Aug. 2009), 63–74. https://doi.org/10.1145/1594977.1592577
[108] Chuanxiong Guo, Guohan Lu, Helen J Wang, Shuang Yang, Chao Kong, Peng Sun, Wenfei Wu, and Yongguang Zhang. 2010. Secondnet: a data center network virtualization architecture with bandwidth guarantees. In *Proceedings of the 6th International COnference*. ACM, 15.
[109] Abhishek Gupta, Paolo Faraboschi, Filippo Gioachin, Laxmikant V Kale, Richard Kaufmann, Bu-Sung Lee, Verdi March, Dejan Milojicic, and Chun Hui Suen. 2016. Evaluating and improving the performance and scheduling of HPC applications in cloud. *IEEE Transactions on Cloud Computing* 4, 3 (2016), 307–321.
[110] Hakan Hacigümüş, Bala Iyer, Chen Li, and Sharad Mehrotra. 2002. Executing SQL over encrypted data in the database-service-provider model. In *2002 ACM SIGMOD int. conf. on Management of data*. ACM, 216–227.
[111] Abdul Hameed, Alireza Khoshkbarforoushha, Rajiv Ranjan, Prem Prakash Jayaraman, Joanna Kolodziej, Pavan Balaji, Sherali Zeadally, Qutaibah Marwan Malluhi, Nikos Tziritas, Abhinav Vishnu, Samee U. Khan, and Albert Zomaya. 2016. A Survey and Taxonomy on Energy Efficient Resource Allocation Techniques for Cloud Computing Systems. *Computing* 98, 7 (July 2016), 751–774.
[112] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee. 2015. Network function virtualization: Challenges and opportunities for innovations. *IEEE Communications Magazine* 53, 2 (Feb 2015), 90–97. https://doi.org/10.1109/MCOM.2015.7045396
[113] Yi Han, Tansu Alpcan, Jeffrey Chan, Christopher Leckie, and Benjamin IP Rubinstein. 2016. A game theoretical approach to defend against co-resident attacks in cloud computing: Preventing co-residence using semi-supervised learning. *IEEE Transactions on Information Forensics and Security* 11, 3 (2016), 556–570.
[114] Yi Han, Jeffrey Chan, Tansu Alpcan, and Christopher Leckie. 2017. Using virtual machine allocation policies to defend against co-resident attacks in cloud computing. *IEEE Tran. on Dependable and Secure Computing* 14, 1 (2017), 95–108.
[115] Tyler Harter, Brandon Salmon, Rose Liu, Andrea C Arpaci-Dusseau, and Remzi H Arpaci-Dusseau. 2016. Slacker: Fast Distribution with Lazy Docker Containers.. In *FAST*, Vol. 16. 181–195.
[116] Brandon Heller, Srinivasan Seetharaman, Priya Mahadevan, Yiannis Yiakoumis, Puneet Sharma, Sujata Banerjee, and Nick McKeown. 2010. ElasticTree: Saving Energy in Data Center Networks. In *Nsdi*, Vol. 10. 249–264.
[117] Scott Hendrickson, Stephen Sturdevant, Tyler Harter, Venkateshwaran Venkataramani, Andrea C Arpaci-Dusseau, and Remzi H Arpaci-Dusseau. 2016. Serverless computation with openlambda. *Elastic* 60 (2016), 80.
[118] Benjamin Hindman, Andy Konwinski, Matei Zaharia, Ali Ghodsi, Anthony D Joseph, Randy H Katz, Scott Shenker, and Ion Stoica. 2011. Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center. In *NSDI*, Vol. 11. 22–22.
[119] Chi-Yao Hong, Srikanth Kandula, Ratul Mahajan, Ming Zhang, Vijay Gill, Mohan Nanduri, and Roger Wattenhofer. 2013. Achieving High Utilization with Software-driven WAN. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM (SIGCOMM '13)*. ACM, New York, NY, USA, 15–26. https://doi.org/10.1145/2486001.2486012
[120] Qian Huang. 2014. Development of a SaaS application probe to the physical properties of the Earth's interior: An attempt at moving HPC to the cloud. *Computers & Geosciences* 70 (2014), 147–153.

[121] Eduardo Huedo, Rubén S Montero, Rafael Moreno, Ignacio M Llorente, Anna Levin, and Philippe Massonet. 2017. Interoperable Federated Cloud Networking. *IEEE Internet Computing* 21, 5 (2017), 54–59.
[122] IDC. 2017. Worldwide semiannual big data and analytics spending guide. http://www.idc.com/getdoc.jsp?containerId=prUS42321417. (Feb 2017). [Last visited on 18th May 2018].
[123] IDG Enterprise. 2016. 2016 IDG Enterprise Cloud Computing Survey. https://www.idgenterprise.com/resource/research/2016-idg-enterprise-cloud-computing-survey/. (2016). [Last visited on 18th May 2018].
[124] IEEE. 2017. IEEE Rebooting Computing. https://rebootingcomputing.ieee.org/. (2017). [Last visited on 18th May 2018].
[125] Shigeru Imai, Thomas Chestna, and Carlos A Varela. 2012. Elastic scalable cloud computing using application-level migration. In *Utility and Cloud Computing (UCC), 2012 IEEE Fifth International Conference on*. IEEE, 91–98.
[126] Shigeru Imai, Thomas Chestna, and Carlos A Varela. 2013. Accurate resource prediction for hybrid iaas clouds using workload-tailored elastic compute units. In *Utility and Cloud Computing (UCC), 2013 IEEE/ACM 6th International Conference on*. IEEE, 171–178.
[127] Shigeru Imai, Thomas Chestna, and Carlos A. Varela. 2013. Accurate Resource Prediction for Hybrid IaaS Clouds Using Workload-Tailored Elastic Compute Units. In *6th IEEE/ACM International Conference on Utility and Cloud Computing (UCC 2013)*. Dresden, Germany.
[128] Shigeru Imai, Pratik Patel, and Carlos A Varela. 2016. Developing Elastic Software for the Cloud. *Encyclopedia on Cloud Computing* (2016).
[129] Shigeru Imai, Stacy Patterson, and Carlos A Varela. 2017. Maximum Sustainable Throughput Prediction for Data Stream Processing over Public Clouds. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE Press, 504–513.
[130] Shigeru Imai, Stacy Patterson, and Carlos A. Varela. 2018. Uncertainty-Aware Elastic Virtual Machine Scheduling for Stream Processing Systems. In *18th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2018)*. Washington, DC.
[131] Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, et al. 2013. B4: Experience with a globally-deployed software defined WAN. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 3–14.
[132] Bahman Javadi, Jemal Abawajy, and Rajkumar Buyya. 2012. Failure-aware resource provisioning for hybrid Cloud infrastructure. *Journal of parallel and distributed computing* 72, 10 (2012), 1318–1331.
[133] Barkha Javed, Peter Bloodsworth, Raihan Ur Rasool, Kamran Munir, and Omer Rana. 2016. Cloud market maker: An automated dynamic pricing marketplace for cloud users. *Future Generation Computer Systems* 54 (2016), 52–67.
[134] Brendan Jennings and Rolf Stadler. 2015. Resource management in clouds: Survey and research challenges. *Journal of Network and Systems Management* 23, 3 (2015), 567–619.
[135] Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. 2017. In-datacenter performance analysis of a tensor processing unit. *arXiv preprint arXiv:1704.04760* (2017).
[136] Christoforos Kachris, Dimitrios Soudris, Georgi Gaydadjiev, Huy-Nam Nguyen, Dimitrios S Nikolopoulos, Angelos Bilas, Neil Morgan, Christos Strydis, et al. 2016. The VINEYARD Approach: Versatile, Integrated, Accelerator-Based, Heterogeneous Data Centres. In *International Symposium on Applied Reconfigurable Computing*. Springer, 3–13.
[137] Yiping Kang, Johann Hauswald, Cao Gao, Austin Rovinski, Trevor Mudge, Jason Mars, and Lingjia Tang. 2017. Neurosurgeon: Collaborative Intelligence Between the Cloud and Mobile Edge. In *Proceedings of the Twenty-Second International Conference on Architectural Support for Programming Languages and Operating Systems*. ACM, 615–629.
[138] S. Kannan, A. Gavrilovska, V. Gupta, and K. Schwan. 2017. HeteroOS - OS design for heterogeneous memory management in datacenter. In *ACM/IEEE 44th Annual International Symposium on Computer Architecture*. 521–534.
[139] James M Kaplan, William Forrest, and Noah Kindler. 2008. *Revolutionizing data center energy efficiency*. Technical Report. Technical report, McKinsey & Company.
[140] Jeffrey O Kephart and David M Chess. 2003. The vision of autonomic computing. *Computer* 36, 1 (2003), 41–50.
[141] Atefeh Khosravi and Rajkumar Buyya. 2017. Energy and Carbon Footprint-Aware Management of Geo-Distributed Cloud Data Centers: A Taxonomy, State of the Art. *Advancing Cloud Database Systems and Capacity Planning With Dynamic Applications* (2017), 27.
[142] Mariam Kiran, Peter Murphy, Inder Monga, Jon Dugan, and Sartaj Singh Baveja. 2015. Lambda architecture for cost-effective batch and speed big data processing. In *IEEE Intl Conf. on Big Data*. IEEE, 2785–2792.
[143] Diego Kreutz, Fernando MV Ramos, Paulo Esteves Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. 2015. Software-defined networking: A comprehensive survey. *Proc. IEEE* 103, 1 (2015), 14–76.
[144] Kubernetes. 2018. Kubernetes - Production-Grade Container Orchestration. https://kubernetes.io/. (2018). [Last visited on 18th May 2018].

[145] Alok Gautam Kumbhare, Yogesh Simmhan, Marc Frincu, and Viktor K Prasanna. 2015. Reactive resource provisioning heuristics for dynamic dataflows on cloud infrastructure. *IEEE Transactions on Cloud Computing* 3, 2 (2015), 105–118.

[146] Raghavendra Kune, Pramod Kumar Konugurthi, Arun Agarwal, Raghavendra Rao Chillarige, and Rajkumar Buyya. 2016. The anatomy of big data computing. *Software: Practice and Experience* 46, 1 (2016), 79–105.

[147] Tung-Wei Kuo, Bang-Heng Liou, Kate Ching-Ju Lin, and Ming-Jer Tsai. 2016. Deploying chains of virtual network functions: On the relation between link and server usage. In *Computer Communications, IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on*. IEEE, 1–9.

[148] Horacio Andrés Lagar-Cavilla, Joseph Andrew Whitney, Adin Matthew Scannell, Philip Patchin, Stephen M Rumble, Eyal De Lara, Michael Brudno, and Mahadev Satyanarayanan. 2009. SnowFlock: rapid virtual machine cloning for cloud computing. In *Proceedings of the 4th ACM European conference on Computer systems*. ACM, 1–12.

[149] Edward A Lee, Björn Hartmann, John Kubiatowicz, Tajana Simunic Rosing, John Wawrzynek, David Wessel, Jan Rabaey, Kris Pister, Alberto Sangiovanni-Vincentelli, Sanjit A Seshia, et al. 2014. The swarm at the edge of the cloud. *IEEE Design & Test* 31, 3 (2014), 8–20.

[150] Guyue Liu and Timothy Wood. 2015. Cloud-scale application performance monitoring with SDN and NFV. In *Cloud Engineering (IC2E), 2015 IEEE International Conference on*. IEEE, 440–445.

[151] Zhenhua Liu, Minghong Lin, Adam Wierman, Steven Low, and Lachlan LH Andrew. 2015. Greening geographical load balancing. *IEEE/ACM Transactions on Networking (TON)* 23, 2 (2015), 657–671.

[152] Mohan Liyanage, Chii Chang, and Satish Narayana Srirama. 2016. mePaaS: mobile-embedded platform as a service for distributing fog computing to edge nodes. In *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2016 17th International Conference on*. IEEE, 73–80.

[153] Raquel V Lopes and Daniel Menascé. 2016. A taxonomy of job scheduling on distributed computing systems. *IEEE Transactions on Parallel and Distributed Systems* 27, 12 (2016), 3412–3428.

[154] Priya Mahadevan, Puneet Sharma, Sujata Banerjee, and Parthasarathy Ranganathan. 2009. A power benchmarking framework for network devices. *NETWORKING 2009* (2009), 795–808.

[155] Redowan Mahmud, Satish Narayana Srirama, Kotagiri Ramamohanarao, and Rajkumar Buyya. 2018. Quality of Experience (QoE)-aware Placement of Applications in Fog Computing Environments. *J. Parallel and Distrib. Comput.* (2018).

[156] Maciej Malawski, Gideon Juve, Ewa Deelman, and Jarek Nabrzyski. 2015. Algorithms for cost-and deadline-constrained provisioning for scientific workflow ensembles in IaaS clouds. *Future Generation Computer Systems* 48 (2015), 1–18.

[157] Zoltán Ádám Mann. 2015. Allocation of virtual machines in cloud data centers-a survey of problem models and optimization algorithms. *ACM Computing Surveys (CSUR)* 48, 1 (2015), 11.

[158] Sunilkumar S Manvi and Gopal Krishna Shyam. 2014. Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. *Journal of Network and Computer Applications* 41 (2014), 424–440.

[159] Garrett McGrath and Paul R Brenner. 2017. Serverless Computing: Design, Implementation, and Performance. In *Distributed Computing Systems Workshops (ICDCSW), 2017 IEEE 37th International Conference on*. IEEE, 405–410.

[160] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. 2008. OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review* 38, 2 (2008), 69–74.

[161] A. M. Medhat, T. Taleb, A. Elmangoush, G. A. Carella, S. Covaci, and T. Magedanz. 2017. Service Function Chaining in Next Generation Networks: State of the Art and Research Challenges. *IEEE Communications Magazine* 55, 2 (February 2017), 216–223. https://doi.org/10.1109/MCOM.2016.1600219RP

[162] Farahd Mehdipour, Bahman Javadi, and Aniket Mahanti. 2016. FOG-Engine: Towards big data analytics in the fog. In *Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 2016 IEEE 14th Intl C*. IEEE, 640–646.

[163] Dirk Merkel. 2014. Docker: lightweight linux containers for consistent development and deployment. *Linux Journal* 2014, 239 (2014), 2.

[164] Rafael Moreno-Vozmediano, Rubén S Montero, and Ignacio M Llorente. 2012. Iaas cloud architecture: From virtualized datacenters to federated cloud infrastructures. *Computer* 45, 12 (2012), 65–72.

[165] Kiran-Kumar Muniswamy-Reddy and Margo Seltzer. 2010. Provenance as first class cloud data. *ACM SIGOPS Operating Systems Review* 43, 4 (2010), 11–16.

[166] Rekha Nachiappan, Bahman Javadi, Rodrigo Calherios, and Kenan Matawie. 2017. Cloud Storage Reliability for Big Data Applications: A State of the Art Survey. *Journal of Network and Computer Applications* (2017).

[167] Thomas D Nadeau and Ken Gray. 2013. *SDN: Software Defined Networks: An Authoritative Review of Network Programmability Technologies.* " O'Reilly Media, Inc.".

[168] Yucen Nan, Wei Li, Wei Bao, Flavia C Delicato, Paulo F Pires, and Albert Y Zomaya. 2016. Cost-effective processing for Delay-sensitive applications in Cloud of Things systems. In *Network Computing and Applications (NCA), 2016 IEEE*

15th International Symposium on. IEEE, 162–169.
[169] Muhammad Naveed, Seny Kamara, and Charles V Wright. 2015. Inference attacks on property-preserving encrypted databases. In *22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 644–655.
[170] NetApp. 2018. NetApp Private Storage for Cloud. https://cloud.netapp.com/netapp-private-storage. (2018). [Last visited on 18th May 2018].
[171] Marco A. S. Netto, Rodrigo N. Calheiros, Eduardo R. Rodrigues, Renato L. F. Cunha, and Rajkumar Buyya. 2018. HPC Cloud for Scientific and Business Applications: Taxonomy, Vision, and Research Challenges. *Comput. Surveys* 51, 1, Article 8 (Jan. 2018), 29 pages.
[172] Radhika Niranjan Mysore, Andreas Pamboris, Nathan Farrington, Nelson Huang, Pardis Miri, Sivasankar Radhakrishnan, Vikram Subramanya, and Amin Vahdat. 2009. PortLand: A Scalable Fault-tolerant Layer 2 Data Center Network Fabric. *SIGCOMM Comput. Commun. Rev.* 39, 4 (Aug. 2009), 39–50. https://doi.org/10.1145/1594977.1592575
[173] Elisabetta Di Nitto, Peter Matthews, Dana Petcu, and Arnor Solberg. 2017. Model-Driven Development and Operation of Multi-Cloud Applications: The MODAClouds Approach. (2017).
[174] Open Networking Foundation. 2017. Software-Defined Networking (SDN) Definition. https://www.opennetworking.org/sdn-resources/sdn-definition. (2017). [Last visited on 18th May 2018].
[175] OpenFog Consortium. 2018. https://www.openfogconsortium.org/. (2018). [Last visited on 18th May 2018].
[176] Claus Pahl and Brian Lee. 2015. Containers and clusters for edge cloud architectures–a technology review. In *Future Internet of Things and Cloud (FiCloud), 2015 3rd International Conference on*. IEEE, 379–386.
[177] Barbara Pernici, Marco Aiello, Jan vom Brocke, Brian Donnellan, Erol Gelenbe, and Mike Kretsis. 2012. What IS Can Do for Environmental Sustainability: A Report from CAiSE'11 Panel on Green and Sustainable IS. *CAIS* 30 (2012), 18.
[178] Jorge E Pezoa and Majeed M Hayat. 2012. Performance and reliability of non-markovian heterogeneous distributed computing systems. *IEEE Transactions on Parallel and Distributed Systems* 23, 7 (2012), 1288–1301.
[179] Chuan Pham, Nguyen H Tran, Shaolei Ren, Walid Saad, and Choong Seon Hong. 2017. Traffic-aware and Energy-efficient vNF Placement for Service Chaining: Joint Sampling and Matching Approach. *IEEE Transactions on Services Computing* (2017).
[180] Raluca Ada Popa, Catherine Redfield, Nickolai Zeldovich, and Hari Balakrishnan. 2011. CryptDB: protecting confidentiality with encrypted query processing. In *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*. ACM, 85–100.
[181] Andrew Putnam, Adrian M Caulfield, Eric S Chung, Derek Chiou, Kypros Constantinides, John Demme, Hadi Esmaeilzadeh, Jeremy Fowers, Gopi Prashanth Gopal, Jan Gray, et al. 2014. A reconfigurable fabric for accelerating large-scale datacenter services. In *2014 ACM/IEEE 41st Int. Symp. onComputer Architecture (ISCA)*. IEEE, 13–24.
[182] M Rajkumar, Anil Kumar Pole, Vittalraya Shenoy Adige, and Prabal Mahanta. 2016. DevOps culture and its impact on cloud delivery and software development. In *Int. Conf. on Advances in Computing, Communication, & Automation (ICACCA)*. IEEE.
[183] Mike Roberts. 2016. Serverless Architectures. https://martinfowler.com/articles/serverless.html. (2016). [Last visited on 18th May 2018].
[184] Benny Rochwerger, David Breitgand, Eliezer Levy, Alex Galis, Kenneth Nagin, Ignacio Martín Llorente, Rubén Montero, Yaron Wolfsthal, Erik Elmroth, Juan Caceres, et al. 2009. The reservoir model and architecture for open federated cloud computing. *IBM Journal of Research and Development* 53, 4 (2009), 4–1.
[185] Bowen Ruan, Hang Huang, Song Wu, and Hai Jin. 2016. A Performance Study of Containers in Cloud Environment. In *Advances in Services Computing: 10th Asia-Pacific Services Computing Conference, APSCC 2016, Zhangjiajie, China, November 16-18, 2016, Proceedings 10*. Springer, 343–356.
[186] Faiza Samreen, Yehia Elkhatib, Matthew Rowe, and Gordon S Blair. 2016. Daleel: Simplifying cloud instance selection using machine learning. In *Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP*. IEEE, 557–563.
[187] Eduardo Felipe Zambom Santana, Ana Paula Chaves, Marco Aurelio Gerosa, Fabio Kon, and Dejan Milojicic. 2016. Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture. *arXiv preprint arXiv:1609.08089* (2016).
[188] Mahadev Satyanarayanan, Pieter Simoens, Yu Xiao, Padmanabhan Pillai, Zhuo Chen, Kiryong Ha, Wenlu Hu, and Brandon Amos. 2015. Edge analytics in the internet of things. *IEEE Pervasive Computing* 14, 2 (2015), 24–31.
[189] Prabodini Semasinghe, Setareh Maghsudi, and Ekram Hossain. 2017. Game Theoretic Mechanisms for Resource Management in Massive Wireless IoT Systems. *IEEE Communications Magazine* 55, 2 (2017), 121–127.
[190] Arash Shaghaghi, Mohamed Ali Kaafar, and Sanjay Jha. 2017. WedgeTail: An Intrusion Prevention System for the Data Plane of Software Defined Networks. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*. ACM, 849–861.
[191] Yogesh Sharma, Bahman Javadi, Weisheng Si, and Daniel Sun. 2016. Reliability and energy efficiency in cloud computing systems: Survey and taxonomy. *Journal of Network and Computer Applications* 74 (2016), 66–85.
[192] Weisong Shi and Schahram Dustdar. 2016. The promise of edge computing. *Computer* 49, 5 (2016), 78–81.

[193] Junaid Shuja, Raja Wasim Ahmad, Abdullah Gani, Abdelmuttlib Ibrahim Abdalla Ahmed, Aisha Siddiqa, Kashif Nisar, Samee U Khan, and Albert Y Zomaya. 2017. Greening emerging IT technologies: techniques and practices. *Journal of Internet Services and Applications* 8, 1 (2017), 9.

[194] Sukhpal Singh and Inderveer Chana. 2016. QoS-aware autonomic resource management in cloud computing: a systematic review. *ACM Computing Surveys (CSUR)* 48, 3 (2016), 42.

[195] Mukesh Singhal, Santosh Chandrasekhar, Tingjian Ge, Ravi Sandhu, Ram Krishnan, Gail-Joon Ahn, and Elisa Bertino. 2013. Collaboration in multicloud computing environments: Framework and security issues. *Computer* 46, 2 (2013), 76–84.

[196] Stephen Soltesz, Herbert Pötzl, Marc E Fiuczynski, Andy Bavier, and Larry Peterson. 2007. Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors. In *ACM SIGOPS Operating Systems Review*, Vol. 41. ACM, 275–287.

[197] Gaurav Somani, Manoj Singh Gaur, Dheeraj Sanghi, Mauro Conti, and Rajkumar Buyya. 2017. DDoS attacks in cloud computing: issues, taxonomy, and future directions. *Computer Communications* 107 (2017), 30–48.

[198] Sander Soo, Chii Chang, Seng W Loke, and Satish Narayana Srirama. 2017. Proactive Mobile Fog Computing using Work Stealing: Data Processing at the Edge. *International Journal of Mobile Computing and Multimedia Communications (IJMCMC)* 8, 4 (2017), 1–19.

[199] Borja Sotomayor, Rubén S Montero, Ignacio M Llorente, and Ian Foster. 2009. Virtual infrastructure management in private and hybrid clouds. *IEEE Internet computing* 13, 5 (2009).

[200] Josef Spillner. 2017. Snafu: Function-as-a-Service (FaaS) Runtime Design and Implementation. *arXiv preprint arXiv:1703.07562* (2017).

[201] Satish Narayana Srirama. 2017. Mobile web and cloud services enabling Internet of Things. *CSI transactions on ICT* 5, 1 (2017), 109–117.

[202] Satish Narayana Srirama and Alireza Ostovar. 2014. Optimal resource provisioning for scaling enterprise applications on the cloud. In *6th International Conference on Cloud Computing Technology and Science (CloudCom)*. IEEE, 262–271.

[203] Brian Stanton, Mary Theofanos, and Karuna P Joshi. 2015. Framework for Cloud Usability. In *International Conference on Human Aspects of Information Security, Privacy, and Trust*. Springer, 664–671.

[204] Brian Stein and Alan Morrison. 2014. The enterprise data lake: Better integration and deeper analytics. *PwC Technology Forecast: Rethinking integration* 1 (2014), 1–9.

[205] Ivan Stojmenovic, Sheng Wen, Xinyi Huang, and Hao Luan. 2016. An overview of fog computing and its security issues. *Concurrency and Computation: Practice and Experience* 28, 10 (2016), 2991–3005.

[206] Melanie Swan. 2015. *Blockchain: Blueprint for a new economy.* " O'Reilly Media, Inc.".

[207] Zahir Tari, Xun Yi, Uthpala S Premarathne, Peter Bertok, and Ibrahim Khalil. 2015. Security and privacy in cloud computing: vision, trends, and challenges. *IEEE Cloud Computing* 2, 2 (2015), 30–38.

[208] The Linux Foundation. 2018. EdgeX Foundry - The Open Interop Platform for the IoT Edge. https://www.edgexfoundry.org/. (2018). [Last visited on 18th May 2018].

[209] Adel Nadjaran Toosi, Rodrigo N Calheiros, and Rajkumar Buyya. 2014. Interconnected cloud computing environments: Challenges, taxonomy, and survey. *ACM Computing Surveys (CSUR)* 47, 1 (2014), 7.

[210] Deepak K Tosh, Sachin Shetty, Xueping Liang, Charles A Kamhoua, Kevin A Kwiat, and Laurent Njilla. 2017. Security implications of blockchain cloud with analysis of block withholding attack. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE Press, 458–467.

[211] Amin Vahdat, David Clark, and Jennifer Rexford. 2015. A purpose-built global network: Google's move to SDN. *Queue* 13, 8 (2015), 100.

[212] Dana Van Aken, Andrew Pavlo, Geoffrey J Gordon, and Bohan Zhang. 2017. Automatic Database Management System Tuning Through Large-scale Machine Learning. In *Proceedings of the 2017 ACM International Conference on Management of Data*. ACM, 1009–1024.

[213] Luis M Vaquero and Luis Rodero-Merino. 2014. Finding your way in the fog: Towards a comprehensive definition of fog computing. *ACM SIGCOMM Computer Communication Review* 44, 5 (2014), 27–32.

[214] Carlos Varela and Gul Agha. 2001. Programming dynamically reconfigurable open systems with SALSA. *ACM SIGPLAN Notices* 36, 12 (2001), 20–34.

[215] Carlos A. Varela. 2013. *Programming Distributed Computing Systems: A Foundational Approach.* MIT Press. 314pp pages. http://wcl.cs.rpi.edu/pdcs

[216] Blesson Varghese, Ozgur Akgun, Ian Miguel, Long Thai, and Adam Barker. 2016. Cloud benchmarking for maximising performance of scientific applications. *IEEE Transactions on Cloud Computing* (2016).

[217] Blesson Varghese and Rajkumar Buyya. 2017. Next generation cloud computing: New trends and research directions. *Future Generation Computer Systems* (2017).

[218] Blesson Varghese, Nan Wang, Sakil Barbhuiya, Peter Kilpatrick, and Dimitrios S Nikolopoulos. 2016. Challenges and opportunities in edge computing. In *Smart Cloud (SmartCloud), IEEE International Conference on*. IEEE, 20–26.

[219] Prateeksha Varshney and Yogesh Simmhan. 2017. Demystifying Fog Computing: Characterizing Architectures, Applications and Abstractions. In *International Conference on Fog and Edge Computing (ICFEC)*.
[220] Sabrina De Capitani Di Vimercati, Sara Foresti, Sushil Jajodia, Stefano Paraboschi, and Pierangela Samarati. 2010. Encryption policies for regulating access to outsourced data. *ACM Transactions on Database Systems (TODS)* 35, 2 (2010), 12.
[221] Kashi Venkatesh Vishwanath and Nachiappan Nagappan. 2010. Characterizing cloud computing hardware reliability. In *Proceedings of the 1st ACM symposium on Cloud computing*. ACM, 193–204.
[222] H. Wang and Laks V.S. Lakshmanan. 2006. Efficient Secure Query Evaluation over Encrypted XML Databases. In *Proc. of VLDB*. Seoul, Korea.
[223] Lan Wang, Olivier Brun, and Erol Gelenbe. 2016. Adaptive workload distribution for local and remote clouds. In *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*. IEEE, 003984–003988.
[224] Lan Wang, Olivier Brun, and Erol Gelenbe. 2016. Adaptive workload distribution for local and remote Clouds. In *2016 IEEE International Conference on Systems, Man and Cybernetics (SMC)* (Budapest, Hungary). IEEE, 003984–003988. https://doi.org/10.1109/SMC.2016.7844856
[225] Lan Wang and Erol Gelenbe. 2015. Adaptive dispatching of tasks in the cloud. *IEEE Transactions on Cloud Computing* (2015).
[226] Lan Wang and Erol Gelenbe. 2018. Adaptive dispatching of tasks in the Cloud. *IEEE Transactions on Cloud Computing* 6, 1 (2018), 33Ű45. https://doi.org/10.1109/TCC.2015.2474406
[227] Nan Wang, Blesson Varghese, Michail Matthaiou, and Dimitrios S Nikolopoulos. 2017. ENORM: A Framework For Edge NOde Resource Management. *IEEE Transactions on Services Computing* (2017). https://doi.org/10.1109/TSC.2017.2753775
[228] Shiqiang Wang, Rahul Urgaonkar, Murtaza Zafer, Ting He, Kevin Chan, and Kin K Leung. 2015. Dynamic service migration in mobile edge-clouds. In *IFIP Networking Conference (IFIP Networking), 2015*. IEEE, 1–9.
[229] Kim Weins. 2015. Cloud Computing Trends: 2015 State of the Cloud Survey. https://www.rightscale.com/blog/cloud-industry-insights/cloud-computing-trends-2015-state-cloud-survey. (2015). [Last visited on 18th May 2018].
[230] Wayne Wolf. 2009. Cyber-physical systems. *Computer* 42, 3 (2009), 88–89.
[231] Song Wu, Chao Niu, Jia Rao, Hai Jin, and Xiaohai Dai. 2017. Container-Based Cloud Platform for Mobile Computation Offloading. In *Parallel and Distributed Processing Symposium (IPDPS), 2017 IEEE International*. IEEE, 123–132.
[232] Miguel G Xavier, Marcelo V Neves, Fabio D Rossi, Tiago C Ferreto, Timoteo Lange, and Cesar AF De Rose. 2013. Performance evaluation of container-based virtualization for high performance computing environments. In *Parallel, Distributed and Network-Based Processing (PDP), 2013 21st Euromicro International Conference on*. IEEE, 233–240.
[233] Liang Xiao, Dongjin Xu, Caixia Xie, Narayan B Mandayam, and H Vincent Poor. 2017. Cloud storage defense against advanced persistent threats: A prospect theoretic study. *IEEE Journal on Selected Areas in Communications* 35, 3 (2017), 534–544.
[234] Mengting Yan, Paul Castro, Perry Cheng, and Vatche Ishakian. 2016. Building a Chatbot with Serverless Computing. In *Proceedings of the 1st International Workshop on Mashups of Things and APIs*. ACM, 5.
[235] Qiao Yan, F Richard Yu, Qingxiang Gong, and Jianqiang Li. 2016. Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges. *IEEE Communications Surveys & Tutorials* 18, 1 (2016), 602–622.
[236] Yonghua Yin, Lan Wang, and Erol Gelenbe. 2017. Multi-layer neural networks for quality of service oriented server-state classification in cloud servers. In *2017 Int. Joint Conf. on Neural Networks (IJCNN),*. IEEE, 1623–1627.
[237] A. J. Younge, J. P. Walters, S. Crago, and G. C. Fox. 2014. Evaluating GPU Passthrough in Xen for High Performance Cloud Computing. In *2014 IEEE International Parallel Distributed Processing Symposium Workshops*. 852–859. https://doi.org/10.1109/IPDPSW.2014.97
[238] Matei Zaharia, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauley, Michael J Franklin, Scott Shenker, and Ion Stoica. 2012. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*. USENIX Association, 2–2.
[239] Bowen Zhou, Amir Vahid Dastjerdi, Rodrigo Calheiros, Satish Srirama, and Rajkumar Buyya. 2017. mCloud: A Context-aware offloading framework for heterogeneous mobile cloud. *IEEE Transactions on Services Computing* 10, 5 (2017), 797–810.
[240] Qunzhi Zhou, Yogesh Simmhan, and Viktor Prasanna. 2017. Knowledge-infused and consistent Complex Event Processing over real-time and persistent streams. *Future Generation Computer Systems* 76 (2017), 391–406.
[241] Tianqing Zhu, Gang Li, Wanlei Zhou, and S Yu Philip. 2017. Differentially Private Data Publishing and Analysis: a Survey. *IEEE Transactions on Knowledge and Data Engineering* (2017).

Received November 2017; revised June 2018; accepted July 2018